DM 7,- Sfr 7.- Ös 56.- Februar ´89 Nr. 2

AMIGA

KICK START

COMPUTER MAGAZIN

ES WAR EINMAL

Wilde Drachen & kühne Helden

AT-KARTE

MS-DOS-Dauerpower

TOP-GAMES

Brandheiße Spiele

TRANSPUTER

Die neue Generation



Spielelösungen • Lernsoftware

RAM-Test • Soundfilter • NEWS

AMIGA-NEWS 2/89

Ein Informationsservice von Ihren Distributoren DTM Deutschland und MICROTRON Schweiz

VIZAWRITE POWER DRUCKERANPASSUNG

Große Sprüche sind nicht meine Art. Aber mit welchem Amiga Textprogramm können Sie Ihren Drucker so ausnützen wie mit Vizawrite?
Hier ein paar Beispiele - der Platz reicht bei weitem nicht aus um alle Möglichkeiten aufzuzählen und der redaktionelle Teil gehört sowieso anderen, also...

Vizawrite lädt beim Starten automatisch eine ASCII Datei mit dem Namen vw.konfiguration ein. Diese enthält Voreinstellungen für das Programm und Anweisungen für die Druckeransteuerung.

Für jeden Amiga Font können zwei Steuersequenzen definiert werden, die zu Beginn des neuen Bildschirmfonts und am Ende diese Sequenz an den Drucker senden. Für Fettschrift, Unterstreichen, Kursiv, Hoch- und Tiefstellen können ebenfalls Steuersequenzen definiert werden.

Die Proportionalschrifttabellen für NEC, Epson LQ und HP Laserjet sind bereits im Programm eingebaut. Wer einen anderen Drucker hat, erstellt die passenden Proportionalschrifttabellen in der Konfigurtionsdatei selbst. Sie müssen dazu lediglich die einzelnen Weiten der Buchstaben im 360stel Zoll kennen.

Die Formfeedsequenz, die das Papier aus dem Drucker befördert können Sie mit einer beliebig langen Kette von Zeichen ersetzen. Wozu das? Ihr Laserdrucker oder Drucker mit bidirektionalem Traktor könnte z.B. nach dem Bedrucken der Seite diese wieder zurücktransportieren und einen Rahmen um das Papier drucken. Oder hätten Sie Lust, einen Download-Zeichensatz direkt aus dem Textprogramm an Ihren Drucker zu senden?

Wer hätte nicht schon gerne Zugriff auf alle eingebauten Schriftarten des Druckers (Wozu kauft man sich denn ein so teures Gerät?) ohne sich um Formatierung und Umbruch der verschieden großen Schriften zu kümmern. Natürlich proportional mit Blocksatz! Welches Textprogramm dies macht, brauche ich Ihnen jetzt nicht zu sagen.

Leider muß ich nun meine Ausführungen abbrechen - die Agentur lamentiert über die unnütz beanspruchte Werbefläche. In der nächsten Ausgabe erzähle ich Ihnen etwas über die HQ (High-Quality) Fonts in Vizawrite. Sind Sie gespannt auf die dreifach höhere Fontauflösung.

Ihr V.Write

Alle Preise sind unverbindlich empfohlene Verkaufspreise. Preisänderungen und Irrtümer vorbehalten. Es gelten unsere AGB.

AUTOBOOT mit Fast-File System!

Als Weltneuheit stellt GVP eine Autoboot Software vor, die erstmalig das Booten direkt über eine Fast-File eingerichtete Partition ermöglicht. Fast-File ist eine neue Organsitation der Festplatte, die wesentlich schneller arbeitet und ab Workbench 1.3 verfügbar ist. Das Einrichten einer Partition mit altem Format wie es bisher notwendig war, entfällt jetzt vollständig. Zusätzlich enthält die neue Software eine vollautomatische Installationsroutine in deutscher Sprache, die jeden angeschlossenen Plattentyp automatisch erkennt und dem Anwender beliebige Kombinationen zur Aufteilung ermöglicht. Die neue Software liegt ab sofort jedem GVP-Produkt bei, der Autoboot Kit ist separat erhältlich.

GVP SCSI Hardcard 20 MByte 28 msec. Seagate Drive	1495,- DM
GVP SCSI Hardcard 42 MByte 19/11 msec. Quantum Prodrive	2295,- DM
GVP SCSI Hardcard 84 MByte 19/11 msec. Quantum Prodrive	3295,- DM
GVP SCSI Controller mit 2 MB Ram Option ohne Festplatte	695,- DM
GVP SCSI Autobootkit für SCSI Hardcard und Controller	50,- DM

Ab 40 MByte werden alle Karten mit Quantum Festplatten bestückt, die zweifelsohne in Kennerkreisen als Porsche unter den Platten gelten. Eine außerordentlich schnelle Zugriffszeit von 19 msec. die sich durch den integrierten 64 KByte Read-Ahead Cache bei geschickter Ausnutzung bis auf 11 msec. steigern läßt, das patentierte Autoparksystem (Airlock) und die fast doppelt so hohe Lebensdauer (MTBF 50.000 Stunden) gegenüber herkömmlichen Festplatten sichern Ihnen jahrelanges, zuverlässiges Arbeiten.

Unsere Bestseller aus über 400 Artikeln

Alle AEGIS Produkte lieferbar! Neu im Programm LIGHTS, CAMERA, ACTION, MODELER 3D und DRAW 2000. Mehr darüber in der nächsten Ausgabe.

Deutsche Anleitungen	
AEGIS Audiomaster	29,95 DM
AEGIS Sonix	39,95 DM
Balance of Power	29,95 DM
Calligrapher	29,95 DM
Comicsetter	29,95 DM
Flugsimulator II	29,95 DM
Jet	29,95 DM
Kampfgruppe	29,95 DM

VIZAWRITE DESKTOP — 228,- DM
Die einzigartige deutsche Textverarbeitung für alle die Ihren Amiga professionell nutzen wollen!

QUARTERBACK dtsch. — 128,- DM
Sichert Ihre Daten von der Festplatte in kürzester Zeit. Sehr umfangreiche Auswahlmöglichkeiten über Maussteuerung.

FUNKTION deutsch — 98,- DM
Das Programm für alle Mathematiker, Studenten und Schüler. Zeichnet bis zu 50 Funktionen in wählbaren Koordinatensystemen, leitet ab und druckt oder speichert die Ergebnisse.

DISCovery deutsch — 198,- DM
Der Amiga Disk Monitor mit leistungsstarken Modulen zur Änderung und Reparatur von Disketten. Sehr übersichtliche Datenausgabe, voll Maus- und Menügesteuert. Im Lieferumfang ist ein 160ig seitiges deutsches Referenzwerk mit ausführlichen Erläuterungen zur Amiga DOS Diskettenstruktur.

PAGEFLIPPER deutsch — 77,- DM
Wollten Sie schon einmal eine Trickfilmsequenz mit eigenen Bildern machen? Pageflipper spielt IFF-Bilder nach dem von Ihnen geschriebenen Drehbuch.

PAGESETTER deutsch — 198,- DM
Das meistverkaufte Desktop-Publishing Programm für den Amiga.

PROFESSIONAL PAGE — 695,- DM
Der große Bruder von Pagesetter mit Postscript Druckausgabe.

EASYL für Amiga 500	695,- DM
EASYL für Amiga 1000	795,- DM
EASYL für Amiga 2000	895,- DM

DIGI-VIEW GOLD setzt Maßstäbe!

Als Nachfolger des inzwischen berühmten Digi-View bringt New-Tek jetzt Digi-View Gold auf den Markt. Im Gegensatz zur alten Version ist nun kein Adapter (Gender Changer) mehr notwendig, der Digitizer läßt sich problemlos am Amiga 500/2000 anschliessen.

Durch eine neuartige Speichertechnik wird jeder Pixel mit einem 21 bit Farbwert digitalisiert, das bedeutet 2,1 Millionen Farbmöglichkeiten! Zusätzlich können jetzt bis zu 100.000 verschiedene Farbwerte auf dem Bildschirm dargestellt werde. Umfangreiche Bearbeitungsmöglichkeiten des gespeicherten Bildes runden das Programm ab.

DIGI VIEW GOLD kostet incl. deutschem Handbuch — 398,- DM



EASYL ZEICHENTABLETT FÜR AMIGA 500/1000/2000

Dieses Grafiktablett eignet sich hervorragend zum Zeichnen und Gestalten und arbeitet garantiert mit allen Programmen zusammnen, die sich per Maus steuern lassen. Die Zeichenfläche arbeitet drucksensitiv und kann dadurch mit jedem Zeichengerät benutzt werden. Die Auflösung beträgt 1024x1024 Punkte auf einer übergroßen DIN A4 Zeichenfläche..

KOSTENLOSE INFORMATIONEN

senden wir Ihnen gerne zu allen Neuheiten und wenn Sie meinen, dies sei alles, bekommen Sie noch unsere Gesamtliste mit über 600 Artikeln. Einfach diesen Coupon auf eine Postkarte kleben und an DTM bzw. in der Schweiz an MICROTRON senden.

O Senden Sie mir Ihre Gesamtliste AMIGA Soft- und Hardware.
O Senden Sie mir detaillierte Informationen zu folgenden Produkten:

Alle unsere Produkte erhalten Sie in gutsortierten Fachgeschäften, die wir Ihnen gerne nennen sowie unseren DTM-Computershops in Wiesbaden. Selbstverständlich können Sie alle Produkte auch direkt schriftlich oder telefonisch bei DTM bzw. MICROTRON bestellen.

MICROTRON
COMPUTERPRODUKTE
Postfach 69 Bahnhofstr. 2
Tel. 032 872429 Fax 032 872482
CH-2542 PIETERLEN

Poststraße 25
6200 Wiesbaden-Bierstadt
(06121) 560084
fax (06121) 563643
DTM
Werbung und EDV GmbH

Editorial

Liebe Leser!

Auf einen kürzlich abgedruckten Artikel, in dem ein Hardwarezusatz sehr gelobt wurde, gab es eine ungewöhnliche Resonanz. Ein Anbieter eines Konkurrenzproduktes rief uns wütend (!) an und drohte uns mit Tod und Teufel. Seine Produkte seien wesentlich besser, selbst XYZ und ABC würden sie benutzen. Zum Abschluß wurde noch die deutsche Justiz erwähnt, und daß man diese anrufen würde. Zu den Beschuldigungen und dem allgemeinen Verhalten des Anrufers soll hier einmal öffentlich Stellung genommen werden, da das Verhältnis Medien - Anbieter wohl von öffentlichem Interesse ist, weil Verbindungen zwischen beiden leicht ins Zwielichtige rücken können.

Für Hinweise auf sachliche Unzulänglichkeiten werden wir immer ein offenes Ohr haben. Wurden Fakten falsch wiedergegeben, wird dies auch richtiggestellt. Wird ein Gerät als das beste auf dem Markt bezeichnet, wird dies kein vernünftiger Mensch als absolutes Urteil ansehen. Jeder Test ist mehr oder weniger subjektiv, da sehr viele Einflüsse auf den Tester einwirken. Zudem sind solche Urteile immer, auch wenn es einmal nicht explizit angegeben ist, preisbezogen zu sehen. Natürlich wird ein 10.000,- DM - Gerät andere Leistungsmerkmale bieten als eines für 500,- DM. Zudem findet ein solcher Vergleich immer zwischen den Geräten statt, die man schon einmal in Augenschein genommen hat. Doch über Formulierungen kann man streiten, niemand ist vollkommen.

Kernpunkt ist aber die Art und Weise, wie auf Tests reagiert wird. Wenn ein Hersteller der Meinung ist, daß sein Gerät besser ist, kann er es uns gerne zu einem Test überlassen. Wird aber in einem mehr oder weniger unfreundlichen Ton reagiert und irgendetwas verlangt oder angedroht, wird man das Gefühl der versuchten Einflußnahme auf die Presse nicht los. Wir wollen Ihnen, liebe Leser, Informationen bieten, die möglichst unabhängig von den verschiedensten Interessen durch uns zusammengestellt wurden. Dies ist uns bisher, hoffen wir, gelungen, und das soll auch in Zukunft so bleiben.

chk

INHALT

AKTUELLES

SOFTWARE

HARDWARE

GRUNDLAGEN

Spiele News

In dieser Ausgabe haben wir den Schwerpunkt auf Spiele gelegt, da erfreulicherweise viele gute Programme erschienen sind. So z.B. Dragons Lair, das durch seine zeichentrickähnlichen Animationen glänzt oder Spiele mit so bekannten Namen wie Emmanuelle oder Dungeon Master. Weiterhin stellen wir Ihnen Lösungen zu bekannten Adventures vor, an denen einige sicherlich schon fast verzweifelt sind.

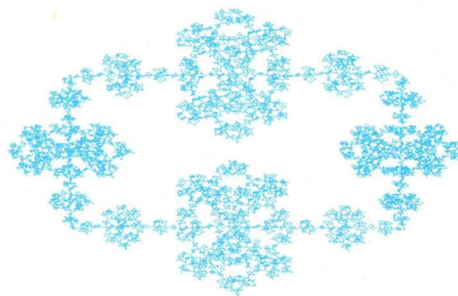


20

Geometrie-Zauber

Auf das Thema Mathematik reagieren viele mit kaltem Schaudern, wird dabei doch manche Erinnerung wach. Doch sobald sie den Computer tangiert, und das auch noch im privaten Bereich, wird manch einer hellhörig. Die Mathematik läßt sich nämlich zu wunderschönen Grafiken umwandeln. Nein, diesmal sind keine Apfelmännchen gemeint, sondern die Koch- und Peanokurven.
Eine Exkursion in die Welt der Computer-Alchemie.

105

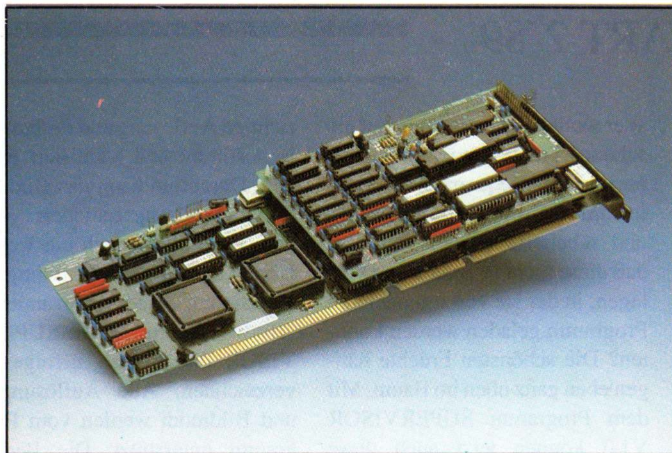


2 '89

Die AT-Karte

Lang, lang ist's her, da kündigte Commodore dieses Wunderwerk der MS-DOS-Welt an. Nach bislang verfügbaren Vorabversionen war Anfang dieses Jahres endlich die Endversion fertig. Neben der Hardware legte Commodore besonderen Wert auf die Software, denn diese hat einiges zu bieten und begeistert sicherlich auch die XT-Karten-Besitzer.

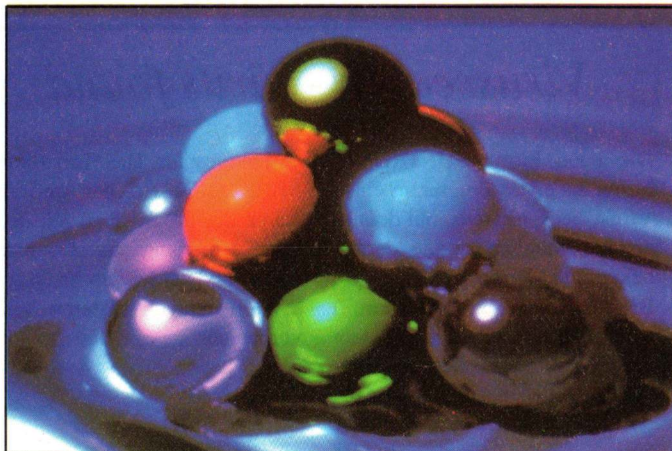
16



Transputer

Wir möchten Ihnen einen kleinen Einblick in die Welt der neuen Prozessoren geben. Zum einen in den 68030, der ja auch bald einen AMIGA antreiben soll, und zum anderen in den T800, weithin als Transputer bekannt. Beide Prozessoren werden in Zukunft die Taktwelt regieren und die Computer noch effizienter arbeiten lassen.

52



LISTING

KICKS FÜR INSIDER

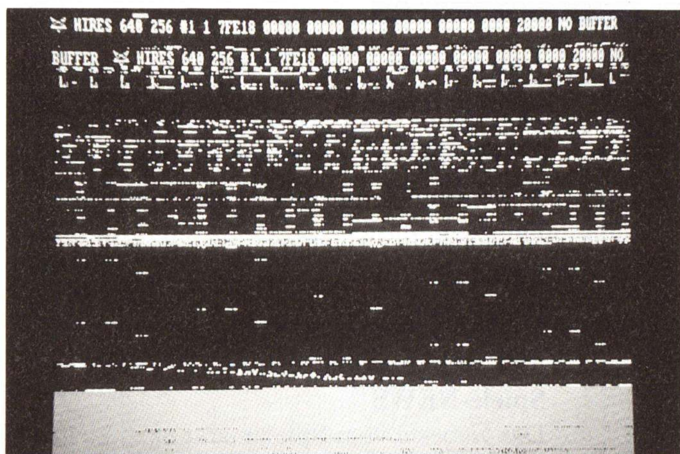
SPIELE

RUBRIKEN

NEWS

KICKSTART 2´89

SUPERVISOR



Der "rohe" Grafikspeicher wird mit Hilfe der Maus und der Tastatur nach Bildern durchsucht.

Wer wollte nicht schon einmal ein schönes Titelbild oder eine interessante Spielsequenz eines Spieles im IFF-Format abspeichern und scheiterte an der Tatsache, daß diese nicht in einer Form vorlagen, in der sie von irgendeinem Programm geladen werden konnten? Die schönsten Früchte hängen eben ganz oben im Baum. Mit dem Programm SUPERVISOR V1.3 können jetzt auch diese Früchte geerntet werden. Es ist in der Lage, beliebige Grafiken im IFF-Format abzuspeichern. Dabei beschreitet das Programm einen eigenwilligen Weg, der sehr effektiv, dafür aber auch nicht ganz einfach ist. Nach einem Reset wird das Programm gestartet, per Hand kann daraufhin der Speicherbereich nach Grafikdaten durchsucht werden. Da Bilder bzw. Grafik nur im CHIPMEM liegen können, wird auch nur dieser Bereich durchsucht. Entdeckt der Anwender die Grafik, müssen die richtige Auflösung und Farbtabelle bestimmt und lokalisiert werden. Daraufhin kann das Bild im IFF-Format abgespeichert werden. Es gehört jedoch ein wenig Übung dazu die Grafik richtig zu selektieren. Gemessen an anderen "Bild-Grabbern" hat der SUPERVISOR die größten Erfolge zu verzeichnen. Alle Auflösungen und Bildmodi werden vom Programm unterstützt. Die Bedienung geschieht sowohl mit der Maus als auch mit der Tastatur. Leider kann es vorkommen, daß das Programm die gewünschten Bilddaten überschreibt und somit zerstört. Zwar ist, um diese unschöne Eigenart zu vermeiden, eine bestimmte Option vorgesehen, sie versagte aber den Dienst. Die Entwickler versprachen jedoch, Abhilfe zu leisten.

Anbieter:
ATLANTIS
Duantstr. 53
5030 Hürth

Neue Vizawrite-Version

Laut Angaben des Herstellers ist ein Vizawrite-Update auf dem Markt. Die neue Version soll nun zwei- bis sechsmal schneller sein, Proportionalschrift ständig unterstützen und farbige IFF-Bilder beim Einladen in bis zu acht verschiedene Graustufen umrechnen. Auch werden nun verschiedene Druckertypen (Laserdrucker, ...) über eine Konfigurationstabelle unterstützt. Registrierte Anwender können sich das Update für 50 DM (40 sFr) bei ihrem Distributor bestellen. Der Ladenverkaufspreis von Vizawrite 2.0 soll bei 228 DM liegen.

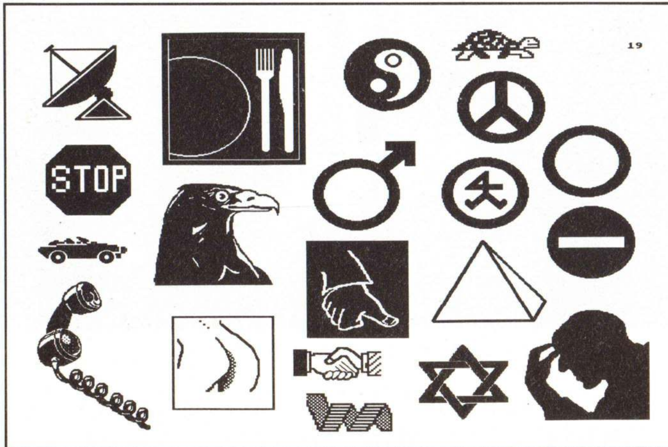
Distributoren:
MICROTRON,
Bahnhofsstr. 2,
CH-2542 Pieterlen DTM
Werbung und EDV GmbH,
Poststr. 25,
D-6200 Wiesbaden

Virussearch - Virus found

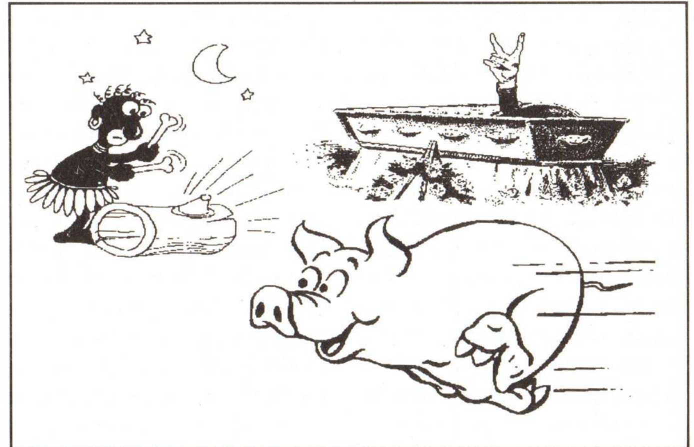
Die neue Firma SUNTRONIC bringt als erstes Programm einen VirusChecker heraus, mit dem es erstmals möglich ist, alle derzeit bekannten Viren (SCA, Byte Bandit, DASA, Byte Warrior,...) sowie Virusprotektoren (System Z,...) auf seinen Disketten ausfindig zu machen und gegebenenfalls zu löschen. Auch der Speicher kann aufatmen, denn auch hier werden Viren gnadenlos verfolgt und aufgespürt. Die Firma liefert SHERLOCK (so der treffende Name des Programms) mit deutscher Anleitung aus und bietet einen Update-Service an. Somit kann man auch späteren Viren vorbeugen.

Anbieter:
SUNTRONIC
Schäfergraben 6
6231 Sulzbach/Ts.

Bilder für Zeitungsmacher



Auf der Diskette sind viele Piktogramme zu finden.

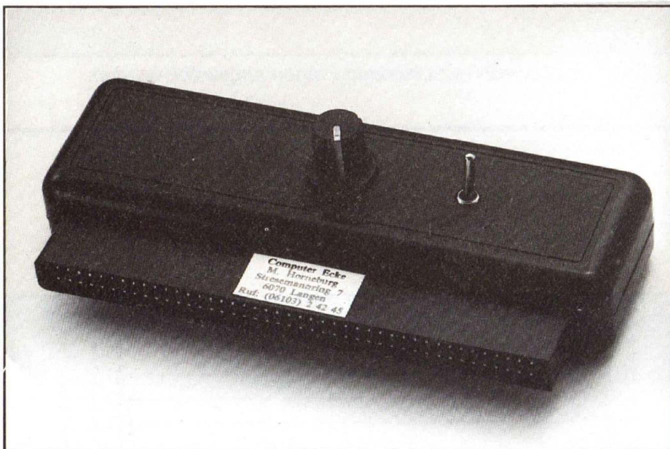

Neben kleineren Bilder finden sich auch komplexere auf der Diskette.

Viele AMIGA-Besitzer, die versuchen, mit ihrem Computer eine zufriedenstellende Zeitung zu erstellen, haben das gleiche Problem: Wie soll man die eintönigen Textwüsten auflockern, wenn man nicht zeichnen kann oder keine Zeit dazu hat? Der Lösung dieses Problems haben sich nun gleich zwei Softwarehersteller gewidmet. Das Kölner Software Studio Plieth bietet eine Diskette mit 400 Kleingraphiken für 79 DM, das Swiss Graphic Design Team gar 6 Disketten (1000 Graphiken) für 99 DM an. Die Bilder beider Anbieter liegen im IFF-Format vor, so daß die Verwendung in dem heimischen DTP-Programm keine Schwierigkeit darstellen sollte. Der Unterschied zwischen den beiden Angeboten liegt in der Art der Graphiken: Die Bilder des SGDT sind gescanned, während die Graphiken aus Köln sozusagen handgemalt sind.

Anbieter:
SoftwareStudioPlieth
Bergisch Gladbacher Str. 696 5000 Köln 80
79.- DM (400 Graphiken)

Swiss Graphic
Design Team
Postfach 201
CH-3072
Ostermundigen 2
99.- DM (1000 Graphiken)

Bremse für AMIGA



Die AMIGA-Bremse findet ihren Bestimmungsort am Expansionport.

Die Firma Computer-Ecke bietet ein neues Zubehör für den AMIGA 500 und 1000 an, mit dem es möglich ist, den AMIGA durch Umlegen eines Schalters anzuhalten (Pausefunktion). Außerdem ist es durch einen Drehregler möglich, die Geschwindigkeit des Rechners stufenlos zu drosseln (z.B. für schwierige Spiele). Die AMIGA 500-Version dieses Moduls (Expansionport) kostet 69.- DM, die 1000er-Version wird zum Preis von 59.- DM angeboten.

Anbieter:
Computer-Ecke
Stresemannring 7
6070 Langen
Tel. 06103/24245

FastBASIC

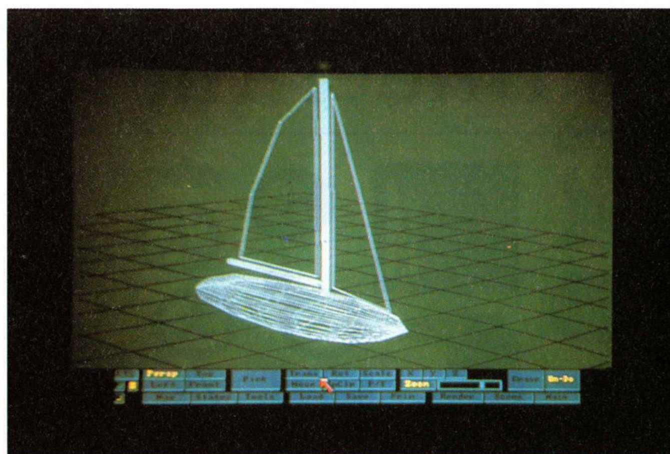
Mittlerweile sind neben dem mitgelieferten AmigaBASIC schon etliche BASIC-Varianten im Umlauf. Mit FastBASIC präsentiert sich eine weitere. FastBASIC ist ein BASIC-Compiler, der nicht nur die BASIC-Grundbefehle beherrscht, sondern mit vielen weiteren
Befehlen, die mitunter von anderen Hochsprachen her bekannt sind, aufwarten kann. Compiliert und gelinkt wird in einem Aufwasch, die Geschwindigkeit ist dabei recht ansehnlich. Die Ablaufgeschwindigkeit der compilierten Porgramme ist ebenfalls sehr schnell und steht anderen Compilern in nichts nach. Wer hofft, bereits geschriebene BASIC-Programme 'mal schnell mit dem FastBASIC-Compiler zu compilieren, dem sei eine Menge Tipparbeit vorausgesagt. FastBASIC ist völlig inkompatibel zu AmigaBASIC, GFA-BASIC oder TRUE BASIC. Ein deutscher Distributor stand zum Zeitpunkt des Redaktionsschlusses noch nicht fest. Wir können nur hoffen, daß die deutsche Version auch ein deutsches Handbuch enthält. Zumindest stellt der FastBASIC-Compiler eine interessante Alternative zu anderen Compilersprachen dar.

Pressenotiz

Zum ersten Mal seit ihrem Bestehen hat die wirtschaftliche Filmförderung des Landes Nordrhein-Westfalen eine Förderung für Computer-Animationen vergeben.
Pictures of Oz-Medienproduktion aus Köln erhielt eine Innovationsförderung des Landes zur Demonstration ihrer Animationsproduktion für den professionellen Medienbereich. Damit ist zum ersten Mal der AMIGA Gegenstand der wirtschaftlichen Unterstützung eines Bundeslandes. Wir meinen: Herzlichen Glückwunsch!

CALIGARI

Das amerikanische Softwarehaus Octree hat in dreijähriger Entwicklungszeit das Programm CALIGARI auf den Markt gebracht, jedoch vorerst nur in Amerika. Das Besondere an diesem enorm leitungsfähigen Programm ist nicht nur der Preis, der für normale AMIGA-Verhältnisse sehr hoch ist und mit ca 4000.- DM (in Worten: viertausend) zu Buche schlägt. Mit diesem Programm soll der Computer-Videomacher ein Werkzeug in die Hand bekommen, das keine Wünsche für Computer-Designer mehr offenläßt. Viele verschiedene Programmteile sind unter dem Programm zusammengefaßt. Vielfältige Objekte können mit Hilfe des Objektmoduls erstellt werden. Das Renderingmodul berechnet die Lichtbrechung, füllt die Flächen und kann noch einiges mehr. Im Script-Modul können komplexe Bewegungsformen definiert werden. Die Länge des Scriptes ist hierbei unbegrenzt. CALIGARI läuft mit mindestens 1 Mbyte RAM. 2-4 Mbyte werden vom Hersteller aber angeraten. Um das Porgramm voll nutzen zu können, sollten des weiteren noch folgende Hardware-Geräte vorhanden sein: Single Frame Controller, Floating Point Accelerator, Broadcast quality Genlock. CALIGARI unterstützt folgende Auflösungen: 736x482 Pixel in 16 Farben oder 362x482 Pixel in 32 bzw. 64 Farben. Eine Anpassung an die deutsche PAL-Norm ist wegen des höheren CHIPMEM-Bedarfs recht schwierig. Für den Heimanwender ist CALIGARI allein schon wegen des hohen Preises wohl kaum konzipiert, eher für Designer und Grafikstudios.



Ein Segelschiff, hier noch als Drahtgittermodell.

ROCK CHALLENGE

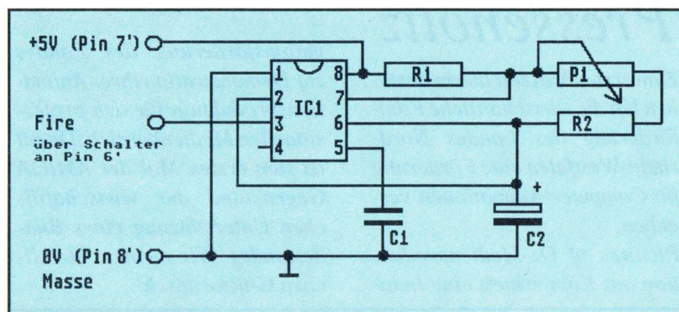
Einen Streifzug durch die Rockmusik kann der AMIGA-Anwender mit ROCK CHALLENGE unternehmen. Realisiert wurde die ganze Sache in Form eines Quizes. In einer Matrix können verschiedene Fragen aus unterschiedlichen Themenbereichen ausgewählt werden. Der Schwierigkeitsgrad ist hierbei gestaffelt. Zur Auswahl stehen mehrere Antworten. Die richtige Antwort ergibt Punkte, wird eine falsche gegeben, besteht eine weitere Chance, jedoch wird dann nur noch die Hälfte der Punkte gezählt. Viel Abwechselung gibt es nicht, auch der ansprechende Sound wird auf die Dauer nervend. Die Steuerung geschieht ausschließlich mit der Maus. Für Rockfreunde, ansonsten öde.

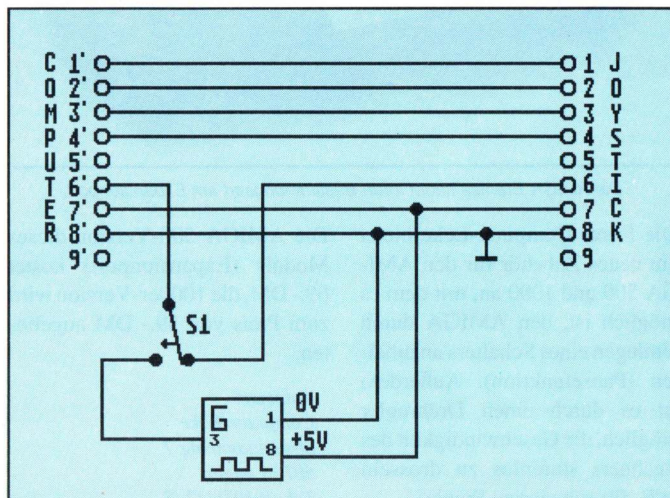


Es können verschiedene Fragen angewählt werden.

Joystick-Dauerfeuer

In der KICKSTART 12/88 stellten wir eine Schaltung zum Erzeugen eines Dauerfeuerimpulses vor. Leider unterliefen dem Zeichner zwei Fehler. Zum einen wurde das IC falsch durchnumeriert, was die meisten unter Ihnen sicher sofort gesehen haben, zum anderen muß die Masse natürlich an Pin 8 geführt werden. Wir bitten dieses Versehen zu entschuldigen und wünschen Ihrem Daumen erholsame Entspannung.





Erfolg durch Perfektion

Jetzt noch schneller

AHD
AMIGA HIGHSPEED HARDDISK
derzeit schnellstes Festplattensystem für Amiga 500/1000 Computer

☐ FAST FILE SYSTEM installierbar ☐ kompatibel zu allen Speichererweiterg. am Expansionp. (z. B. Golem Box) ☐ vorbereitet f. Autobootkickstart (V 1.4) ☐ Aufteilung in mehrere logische Laufwerke leicht möglich ☐ incl. leistungsfähigem Backup-Programm zur Datensicherung auf Diskette ☐ incl. Utilities (z. B. zum Lesen der Preferences von der Harddisk, u.v.m.) ☐ incl. ca. 10 MByte Public Domain Software ☐ automatisches Parken der Schreib-/Leseköpfe (autopark) ☐ 150 Watt Schaltnetzteil eingebaut!! mit Anschluß für Amiga 500 ☐ zusätzliches Laufwerk im gleichen Gehäuse leicht nachrüstbar – PREISVORTEIL!

AHD–MFM DM **998,–**
AHD–RLL DM **1048,–**
AHD–20 (20 MByte) DM **1498,–**
AHD–30 (30 MByte) DM **1798,–**
AHD–40 (40 MByte/40 ms) DM **2198,–**
AHD–40/I (40 MByte/28 ms) DM **2498,–**
AHD–60 (60 MByte) DM **2498,–**

DRIVE EXPANSION BOX
☐ für Diskettenlaufwerke ohne BUS-Durchführung
☐ 3 Diskettenlaufwerke anschließ-, abschaltbar
☐ geeignet für 3,5"- sowie 5,25"-drives
☐ Drive 1 und Drive 2 vertauschbar
DM **79,–**

MIDIFACE
☐ für Amiga 500/1000/2000 Typ bitte angeben
☐ kompatibel zu allen MIDI-Programmen
☐ 1 × MIDI IN, 3 × MIDI OUT, 1 × MIDI OUT/THRU schaltbar
☐ mit Kontrollanzeige für MIDI IN und MIDI OUT
DM **129,–**

Kickstartumschaltung **MK–1**
☐ für Amiga 500 und 2000
☐ kompl. steckbar, kein Löten
Fertiggerät DM **59,–**
Leerplatine DM **39,–**
Software zum Erstellen brennfertiger Files von Ihrer Kickstartversion:
Kickloader DM 39,–
Eprombrennservice für MK–1 DM **39,–**

PAL-Genlock-Interface
☐ geeignet für Amiga 500/1000/2000
☐ getrennte Regelung von Computer und Videosignal (fade-in/fade-out)
☐ FBAS und RGB-Ausgang
☐ Amigamonitor ist als Kontrollbildschirm nutzbar
DM **598,–**

Audio-Digitizer
☐ kompatibel zu allen Digitizerprogrammen
☐ top Qualität
incl. Software DM **79,–**

Epromprogrammiergerät
☐ für Amiga 500/1000/2000
Preis auf Anfrage!

Zusatzlaufwerk **MAD–II+**
☐ 100 % kompatibel zum Original
☐ Anschluß durchgeführt
☐ abschaltbar
DM **298,–**

Echtzeituhr **MCT–1000**
☐ VIRUSGESCHÜTZT
☐ anschlußfertig für Amiga 1000
☐ Betrieb am Expansionport
☐ akkugepuffert
☐ Schreibschutzschalter gegen versehentliches Verstellen
☐ quarzgenau
☐ incl. Steuersoftware zum Einbinden i die Startup-Sequenz
DM **98,–**

512 kByte Speichererweiterung für Am 500
siehe AMIGA MAGAZIN 8/88
☐ erweitert den Arbeitsspeicher auf 1 MByte
☐ akkugepufferte Uhr
☐ komplett abschaltbar
Leerplatine mit Stecker DM **39,–**
Fertiggerät ohne RAMs DM **89,–**
Fertiggerät mit RAMs auf Anfrage

Netzkabel zum Anschluß des Amiga 500 an AHD-Systeme!
DM **49,–**

150W Schaltnetzteil für Amiga 500
DM **298,–**

Abschaltung für Amiga 500-Speichererweiterung
(z. B.: A 501)
DM **29,90**

Centronics-Druckerkabel
für Amiga 500/1000/2000
DM **29,–**

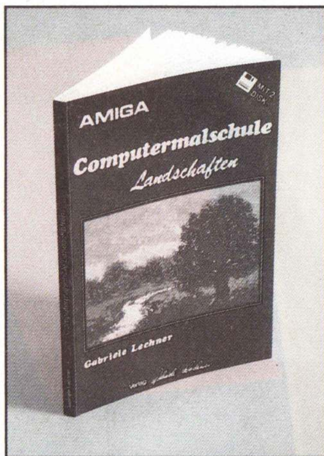
Bestellung und Versand:
telefonisch: **02 08/2 40 47**
per BTX-Mitteilung: **020824049**
oder schriftlich

Versandspesen DM 8,70 per Nachnahme (Inland) oder Vorauskasse, Versand ins Ausland nur gegen Vorauskasse plus DM 10,–

MESSAGE Computer

Message Computer
Thomas Martin/Andreas Gerzen
Stöckmannstr. 78
4200 Oberhausen 1

Österreich: SUETRAK HANDELSGES. M. B. H.
Mitterauen 31 · A-3003 Gablitz · Tel. 0 22 31/21 70



Computermalschule: Landschaften

Gabriele Lechner
Verlag G. Lechner
ca. 160 Seiten
DM 59.- (incl. 2 Disketten)

Sicherlich hat jeder AMIGA-Besitzer ein Malprogramm. Doch nur die wenigsten werden es wirklich ausnutzen. Und dabei ist ein Computer gerade für die ersten Versuche gut zu gebrauchen, erlaubt er es doch, unerwünschte "Pinselstriche" spurlos und einfach wieder verschwinden zu lassen, Bilder zwischenzuspeichern usw. Unter dem Titel "Computermalschule" sind im Verlag G. Lechner bereits mehrere Bücher erschienen. Das neueste Werk dieser Serie ist ganz der Landschaftsmalerei gewidmet.

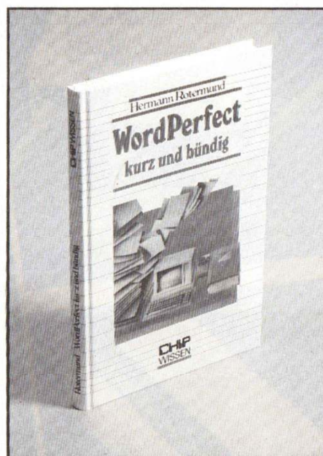
Das Buch versteht sich als eine Einführung in die Landschaftsmalerei. Vorausgesetzt werden nur Kenntnisse im Umgang mit DPaint II und ein wenig Kreativität. Die weiteren "Zutaten" für ein gelungenes Bild sollen während des Durcharbeitens der Übungen erarbeitet werden.
Im ersten Kapitel werden die notwendigen Grundelemente und Techniken der (Landschafts-)Malerei behandelt, z.B. Motivsuche, Bildkomposition, Farben und Tonwerte, Perspektive, Licht und Schatten. Im zweiten Kapitel folgen verschiedene Landschaftselemente wie z.B. Bäume, Wolken, Mauern, Gebäudeteile.
Daraus werden dann im dritten Kapitel Schritt für Schritt komplette Landschaftsbilder erstellt. Begonnen wird mit der einfachen Skizze, die jedoch schon alle gestalterischen Elemente enthalten sollte. Danach wird diese Skizze nach und nach zum fertigen Bild weiterbearbeitet. Besonders hilfreich ist hierbei vor allem, daß sich alle Bilder und Brushes auf den beiden mitgelieferten Disketten befinden. Dadurch ist es auch einem Ungeübten möglich, mit Hilfe der Skizzen eine Zeichnung zu erstellen, denn gerade das Anfertigen einer Skizze bereitet anfangs meist größere Probleme.

Ein Kritikpunkt am Buch ist seine karge Gestaltung. Dies betrifft in erster Linie die Bilder, die mehr schlecht als recht auf s/w konvertiert wurden. Dies sollte jedoch den Leser nicht zu sehr stören, denn schließlich sind alle Bilder auf den beiden beiliegenden Disketten enthalten. Ansonsten kann man der Autorin bescheinigen, daß sie es beherrscht, auf dem Computer stimmungsvolle Landschaftsbilder zu erstellen. Wer also gerne in das Malen von Landschaftbildern eingeführt werden will oder noch einige Anregungen benötigt, dem kann dieses Buch empfohlen werden.

Word Perfect - kurz und bündig

Hermann Rotermund
1988, Chip Wissen
ca. 145 Seiten
DM 30.-

Word Perfect ist die zur Zeit beste Textverarbeitung für den AMIGA. Das liegt zum einen an ihrem klaren Konzept und ihrer weiten Verbreitung auf anderen Rechnersystemen und zum anderen an den vielfältigen Fähigkeiten, die sie besitzt. Doch gerade diese Möglichkeiten stellen selbst bei öfterer Anwendung den Benutzer vor die Frage nach den dazu notwendigen

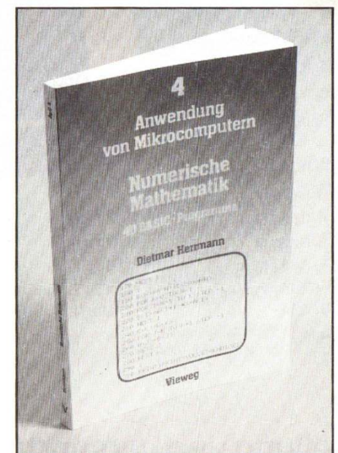


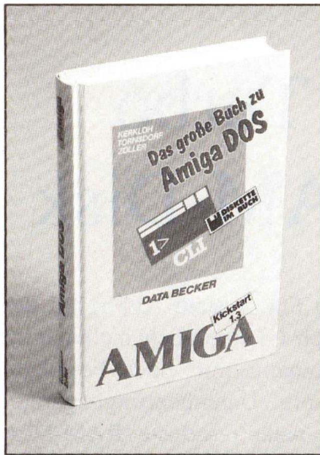
Schritten und Tastenkombinationen. Zwar wird zu Word Perfect eine Tastaturschablone mitgeliefert, aber es werden hier nur die häufigsten Befehle eingesetzt, und auch die eingebaute Help-Funktion ist meistens nicht ausführlich genug. Hier setzt die Buchreihe "kurz und bündig" an. Auf knapp 150 Seiten werden in 11 Kapiteln die verschiedenen Themenbereiche des Programms kurz und bündig vorgestellt. Dabei wurde im Gegensatz zum viel umfangreicheren Handbuch besonderer Wert auf eine übersichtliche, klare Gliederung der einzelnen Aktionen gelegt. Besonders hervorgehoben werden die dazu notwendigen Tastenkombinationen. Behandelte Themen sind neben Grundsächlichem und einer kurzen Einführung die Bereiche Texteingabe, Textformatierung, Makros, Mischfunktion, Rechnen im Text, Fuß- und Endnoten, Inhaltsverzeichnis, Index und Listen.
Dem Buch liegt zwar die Version 4.2 von Word Perfect zugrunde, und für den AMIGA gibt es nur 4.1 (wo man beim IBM-PC doch schon bei 5.0 und der Grafikeinbindung ist!), doch die Unterschiede sind nicht so groß, daß sie ins Gewicht fallen.
Das Buch ist auf jeden Fall eine wertvolle Hilfe beim täglichen Arbeiten mit Word Perfect, denn der Umgang mit dieser leistungsstarken Textverarbeitung wird hiermit wirklich erleichtert.

Anwendungen von Microcomputern

Numerische Mathematik
Dietmar Herrmann
1985, Vieweg

Es gibt viele Anlässe, sich mit numerischer Mathematik auseinandersetzen zu müssen. Zum einen ist das während der Schulzeit und im Studium, zum anderen findet die numerische Mathematik jedoch auch Anwendung in der Praxis (Wofür sollte man sie sonst erlernen?!). Die Programme sind als Basic-Listing angegeben. Dies sollte man jedoch nicht als Abwertung auffassen, denn erstens lassen sie sich leicht auf eine andere Hochsprache umsetzen und dabei auch optimieren, und außerdem gibt es für jeden Algorithmus ein ausführliches Struktogramm, so daß wirklich der Weg zu jeder Hochsprache offensteht. Behandelt werden unter anderem: Funktionsentwicklung nach Fourier, nichtlineare und lineare Gleichungen, Fehlerrechnung, Ausgleichsrechnung, Interpolation, numerische Differentiation und Integration, Eigenwert von Matrizen und partielle Differentialgleichungen.
Die jeweiligen Algorithmen werden kurz hergeleitet, wobei an manchen Stellen sicherlich eine ausführlichere Beschreibung wünschenswert gewesen wäre. Die Programme sind spärlich bis gar nicht dokumentiert. Auch hier wären an der ein oder anderen Stelle ein paar Worte zur Programmrealisierung nicht das Falscheste gewesen.
Alles in allem ist das Buch eine wertvolle Sammlung von Algorithmen, die sich leicht nutzen lassen. Wer weiterführende Erläuterungen benötigt, der sollte sich der Fachliteratur bedienen.





Das große Buch zu AMIGA DOS

Kerkloh, Tronsdorf, Zoller
Data Becker
370 Seiten
ISBN 3-89011-306-0
59,- DM

Das Buch ist an sich keine Neuerscheinung, sondern wurde nur in Bezug auf die Workbench 1.3 überarbeitet. Durch die diversen Änderungen, die ins Betriebssystem des AMIGA eingeflossen sind, hat sich der Inhalt des Buches doch stark verändert, so daß man es als recht aktuell betrachten kann. Das Buch zeichnet sich durch eine gute Strukturierung aus.

Thematisch zusammengefaßt werden die einzelnen Bereiche des DOS erläutert. Das erste Kapitel stellt eine Einführung in die Verwendung des Computers dar. Diesem schließt sich das Kapitel über die CLI-Befehle und ihre Aufgaben an. Das Kapitel ist übersichtlich in vier Abschnitte gegliedert. Zuerst werden die dateiorientierten Befehle erläutert. Nächster Punkt sind die Systemkommandos, denen sich eine Betrachtung der Batchbefehle anschließt.

Was natürlich auch nicht fehlt, ist die Erläuterung der Editoren ED und Edit. Wenn man die Abhandlung des ED noch als sinnvoll erachten kann, stößt die Erläuterung des Edit doch auf Unverständnis. Oder kennen Sie auch nur einen, der diesen Editor benutzt? Ich nicht.

Die Erläuterung der Befehle ist gut gelungen. Neben einer Syntaxangabe findet man eine ausführliche Erklärung der Befehle und ihrer Möglichkeiten. Am Ende jeder Beschreibung findet man die Änderungen, die sich unter DOS 1.3 ergeben.

Drittes Kapitel ist die Erläuterung der Standard-Devices. Dabei wird erläutert, welche Hardware mit den Devices angesprochen wird und welche Besonderheiten bei der Verwendung bestehen. Gut gefallen hat die Erläuterung des Seriell-Devices und seiner Unzulänglichkeiten. Dabei stört es nur ein wenig, daß auf Kapitel verwiesen wird, die nicht vorhanden sind. Bei der Umgestaltung des Buches hat sich wohl etwas verschoben, da man die Abhandlungen unter einer anderen Kapitelnummer schließlich doch findet.

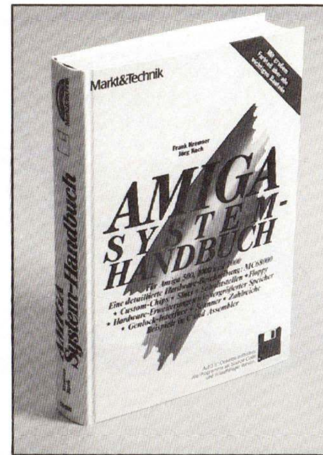
Im nächsten Kapitel werden dann die neuen Befehle und Devices der Workbench erklärt. Nachteilig fällt hier auf, daß man die Angaben wohl aufgrund einer Vorversion des neuen DOS gemacht hat (was bei dem Erscheinungsdatum des Buches auch zwingend ist), wodurch sich einige Unterschiede zu den mir vorliegenden DOS 1.3-Versionen ergaben. Auch wird in diesem Kapitel geschrieben, daß man die resetfeste RAM-Disk formatieren müßte. Wie man zu dieser Erkenntnis kam, ist mir schleierhaft.

Nach einigen Hinweisen zum Arbeiten mit dem CLI folgt im sechsten Kapitel eine ausschweifende Erläuterung des Batch-Prozessings. Dabei wird anhand von Beispielbatches der Sachverhalt recht plastisch geschildert.

Nächster Punkt ist das Multitasking. Was Multitasking ist und die Realisation und Verwendung auf dem AMIGA werden abgesteckt. Anschließend wird auf Interna des DOS eingegangen. Dies bleibt aber oberflächlich und kann nur als Erläuterung für den interessierten Anwender angesehen werden. Im abschließenden Kapitel wird auf selbstgeschriebene CLI-Befehle eingegangen. Einige Befehle werden vorgestellt, die aus der Feder der Buchautoren stammen. Auf die Programmierung wird allerdings nicht eingegangen.

Beigelegt ist eine Diskette, die alle größeren Listings aus dem Buch enthält. 59,- DM erscheinen mir dafür angemessen.

chk



AMIGA System-Handbuch

Kremser, Koch
Markt&Technik
ISBN 3-89090-550-1
420 Seiten
79,- DM

Das Werk beschäftigt sich ausschließlich mit der Hardware der Rechner und ihrer Programmierung. Dabei werden aber nicht nur die Spezialchips und Portbausteine erläutert. Auch grundlegende Sachverhalte werden dargelegt. So beschäftigt sich ein Kapitel mit den verschiedenen Slots des AMIGA und den dort anliegenden Signalen. Dabei wird auch auf die PC-Slots eingegangen. Ein weiterer Abschnitt behandelt die Floppy. Dabei wird sehr genau die Funktionsweise eines Diskettenlaufwerkes beschrieben. Die möglichen Aufzeichnungsverfahren werden erläutert und die Programmierung der floppyspezifischen Chips erklärt.

Im Abschnitt über die Schnittstellen werden auch die Unterschiede der Belegungen bei den verschiedenen AMIGA-Modellen aufgeführt. Den Schnittstellen zur Tastatur und zur Maus wurden jeweils eigene Abschnitte im Buch zugeordnet. Die Funktion der beiden Standardeingabegeräte wird erläutert und die Programmierung der Devices beschrieben.

Ein weiteres Kapitel ist kleinen Bastelanleitungen gewidmet. Digitizer, Scanner, Schalt-Interfaces und sogar ein kleines Genlock werden vorgestellt. Dabei handelt es sich durchweg um kleine und preiswerte Schaltungen, die leicht realisierbar sind. Gesondert werden RAM-Erweiterungen behandelt. Dabei wird auch die Differenz zwischen statischen und dynamischen RAMs und ihrer Verwendung angesprochen.

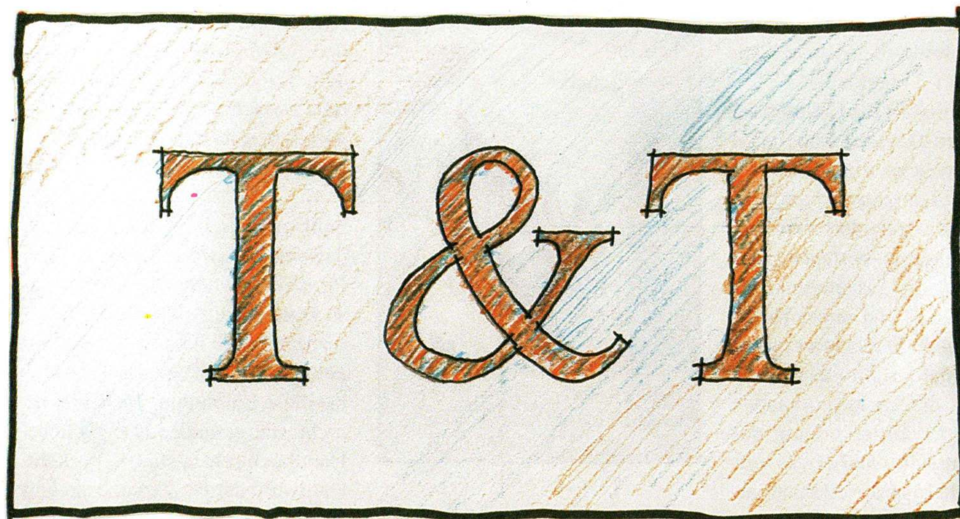
Sehr erfreulich ist die Tatsache, daß sich dieses Buch auch mit den PC-Teilen befaßt. Der AMIGA ist so konzipiert, daß diese Hardwarezusätze sich nahtlos ins System einfügen lassen. In der Literatur hat diese Einbindung bisher leider nicht stattgefunden.Das System-Handbuch geht ausführlich auf die Hardware der PC-Zusätze ein und erfaßt alle wichtigen Teile. Die Angaben wurden größtenteils aus dem Technical Reference Manual übernommen. Da dies die einzige tiefgreifende Abhandlung über die PCs und zudem nicht jedem verfügbar ist, ist dieses 'Abkupfern' durchaus nicht als verwerflich anzusehen. Ebenfalls erläutert ist die zum PC gehörige Janus-Library. Leider wird nicht auf das jdisk.device eingegangen, was wohl in nicht verfügbarem Informationsmaterial seine Begründung hat.

Das letzte Kapitel befaßt sich mit der Expansion-Library, die dazu dient, Hardware-Erweiterungen ins System einzubinden. Mit dem bis hierher gebotenen Wissen kann man dann schon fast seine eigenen Erweiterungskarten konstruieren.

Erwähnenswert ist der Bildteil in der Mitte des Buches. Hier findet man Farbfotos aller Hardwareteile der verschiedenen AMIGA-Modelle, was die Orientierung im Rechner immens erleichtert.

Im Anhang findet man einige nützliche Angaben wie Kartengrößen, Speicherbelegung, Adressen der Hardwareregister oder der Portbausteine. Fast schon obligatorisch ist eine Diskette, die dem Buch beigefügt ist. Darauf findet man alle Beispielprogramme im Source-Code und als lauffähige Programme.

chk



FastMem ?

Dieses nur vom CLI startbare Programm testet, ob eine bestimmte Größe FastMem verfügbar ist. Dadurch läßt sich zum Beispiel feststellen, ob eine Speichererweiterung eingeschalten ist. Ein sinnvoller Einsatz wäre in einer Startup-Sequence denkbar. Wenn weniger als 64K (in unserem Beispiel) ansprechbar sind, wird das Programm mit einer Warnung beendet. So ist es möglich, in einem Batchfile den Start eines Hintergrundprozesses von einem FastMem-Test abhängig zu machen:

```
MemEx                ; unser programm
if NOT WARN          ; fastmem vorhanden?
   runback clock     ; uhr als task laden
endif                ; nein => weiter
```

Hier der Assemblerquellcode des Programmes (SEKA):

```
start:  move.l  $04,a6          ; execbase nach a6
        moveq   #04,d1          ; FastMem
        jsr     -216(a6)        ; AvailMem
        cmp.l   #$10000,d0      ; über 64K frei?
        bgt.s   ja
        moveq   #05,d0          ; returncode in d0
        rts                     ; zurück mit warnung
ja:     moveq   #00,d0          ; keine warnung
        rts
```

(Harald Zacharias)

Turbo RAM-Disk

Das "Fast File System" (FFS) auf der Workbench 1.3 ermöglicht nicht nur im Zusammenhang mit einer Festplatte eine höhere Datenübertragungsrate, auch die neue resetfeste RAM-Disk ("CARD:") kann unter FFS arbeiten. Man muß in der Mountlist bei CARD: lediglich zwei Änderungen einfügen:

```
GlobVec = -1
FileSystem = L:FastFileSystem
```

Danach installiert man das Device CARD: mit dem Befehl "Mount CARD:" und formatiert die RAM-Disk folgendermaßen:

```
Format drive CARD: name Turbo
                    noicons quick
```

Zum Vergleich hier noch die von mir ermittelten Zeiten beim Laden von Amiga-Basic:

2.54 Sekunden mit FFS
3.45 Sekunden ohne FFS

Außerdem hat die RAM-Disk jetzt 5 Prozent mehr Speicherkapazität. Leider hat dieser Tip auch den Nachteil, daß diejenigen, die bereits im Besitz des inoffiziellen Kickstarts 1.3 sind, nicht mehr aus der CARD: booten können.

(Fred Kroll)

Schnelle Punkte in Assembler

Als Zugabe hier noch ein kleines Listing, mit dem es möglich ist, in Assembler Punkte durch direkte Adressierung einer Bitplane zu setzen oder zu löschen. Um einen Punkt zu löschen, muß "*loeschen*" aufgerufen werden, um ihn zu setzen wird - na? - ja, es wird "*setzen*" angesprungen. Der Subroutine müssen folgende Parameter mit auf den Weg gegeben werden:

in D0: die x-Koordinate des Punktes
in D1: die y-Koordinate des Punktes
in A5: den Zeiger auf die Bitplane

Haben Sie die Bitplane nicht mit dem Copper auf den Bildschirm gebracht, sondern einen Screen geöffnet, so bekommen Sie den Zeiger durch Addieren von $c0 zum ScreenPointer:

```
move.l ScreenPointer,a5
move.l $c0(a5),a5
```

Hier aber das Listing:

```
loeschen: move.b #$b5,l+1
setzen:   mulu   #08,d0
          divu   #40,d1
          add.w  d0,d1
          swap   d0
          move.b #07,d2
          sub.b  d0,d2
l:        bset   d2,(a5,d1)
          move.b #$f5,l+1
          rts
```

(Bernd Lorenz)

(Fast) Booten von Festplatte

Wenn Sie Besitzer einer Festplatte sind, werden Sie sich sicher auch ärgern, daß das Booten von Festplatte (noch) nicht möglich ist. Wenn Sie jedoch folgende Tips beachten, wird es Ihnen möglich sein, eben dieses zu tun. Eine kleine Einschränkung gibt es noch - nach dem Einschalten des Computers wird trotzdem noch die Diskette benötigt. So, nun geht's los.

Tippen Sie Listing 1 ab und speichern Sie es unter dem Namen "*s/startup- sequence*" auf Ihrer Festplatte . Dann wird Listing 2 unter demselben Namen auf der Bootdiskette und unter "*s/startup-sequence.card*" auf die Festplatte gesichert. Jetzt benötigen sie nur noch die drei richtigen (möglichst kurzen) Mountlists. Auf Platte kommt die normale auf Workbench 1.3.3 enthaltene *MountList* plus die Definition Ihrer HardDisk (meist dh0:), auf die Bootdiskette kommt nur Listing 3 (bzw. die Definition Ihrer Harddisk), das Sie auch unter "*devs/mountlist.card*" auf die Festplatte speichern müssen.

Listing 1

```
dh0:c/assign c: dh0:c
assign sys: dh0:
assign s: dh0:s
assign c: dh0:c
assign l: dh0:l
assign devs: dh0:devs
assign libs: dh0:libs
system/setmap d
echo "A500/A200 Workbench Disk.  Release 1.3 GAMMA Version 34.5*N"
makedir ram:t
makedir ram:env
assign t: ram:t
assign env: ram:env
path sys:utilities sys:system ram: sys: s: add
failat 11
mount
nil>: card:
if not error
   makedir card:devs
   makedir card:s
   copy disk.info card:
   copy s/startup-sequence.card card:s/startup-sequence
   copy c/monut card:
   copy c/cd card:
   copy devs/mountlist.card card:devs/mountlist
   copy devs/system-configuration card:devs
   copy devs/harddisk.device card:devs
endif
mount newcon:
resident cli l:shell-seg
SYSTEM pure
resident c:execute pure
setclock opt load
copy card:disk.info ram:
wait 3
loadwb -debug
endcli >NIL:
```

(Achim Kämmler)

Listing 2

```
mount dh0:
cd dh0:
dh0:execute
dh0:s/startup-sequence
```

Listing 3

```
DH0:
    Device = harddisk.device
    Unit = 1
    Flags = 0
    Surfaces = 4
    BlocksPerTrack = 26
    Priority = 5
    Reserved = 2
    Interleave = 0
    LowCyl = 0
    HighCyl = 612
    Buffers = 5
    BufMemType = 1
```

FULLSCREEN Teil 3

Leider ist uns im letzten Heft ein Fehler unterlaufen denn es wurden nur die ersten drei Punkte der Anpassung von FULLSCREEN an BIGCLI angegeben. Deshalb folgt nun noch einmal die komplette Schritt-für-Schritt-Anleitung für eine individuelle Anpassung von FULLSCREEN an das jeweilige Datensichtgerät:

1. Starten Sie FULLSCREEN auf einer "unbehandelten" Disk.

2. Booten Sie das System mit dieser Disk neu, um die geänderten Preferences-Einstellungen zu aktivieren.

3. Geben Sie den CLI-Befehl "NewCLI con:0/0/672/274/C"

4. Sollte der untere oder gar rechte Rand dieses neuen CLI-Windows auf Ihrem Bildschirm nur teilweise sichtbar sein, so löschen Sie das neue Window, indem Sie es anklicken und "ENDCLI" eingeben.

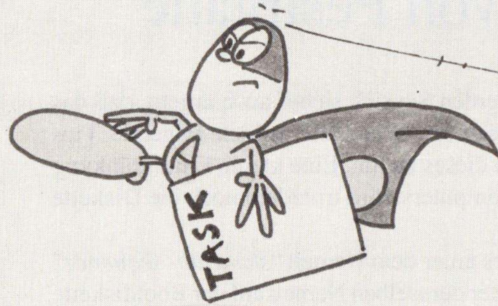
5. Im "alten Window" wiederholen Sie nun den Befehl unter Punkt 3 mit kleineren Werten für die unpassenden x- oder y-Koordinaten des zu öffnenden Windows.

6. Entsprechend dem Ergebnis wiederholen Sie die Punkte 4 und 5 solange, bis Sie Ihre persönliche Idealgröße ermittelt haben.

7. Von der so gefundene y-Größe (Bildhöhe) ziehen Sie den Wert 256 ab; das Ergebnis ist der neue y-Offset, den Sie nur noch in Hexadezimal umrechnen müssen un in der Listingzeile 30 anstelle der "12" entragen. Von einem ebenfalls ermittelten x-Wert ziehen Sie nicht 256, sondern 640 ab, rechnen das Ergebnis ebenfalls in Hex um und tragen es in Zeile 32 anstelle der "20" ein.

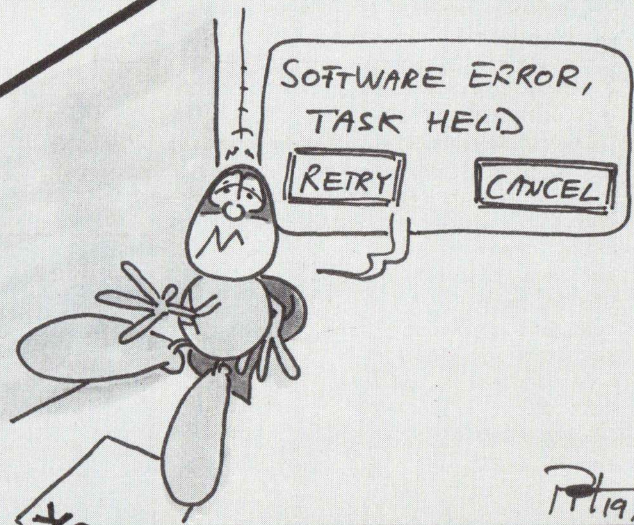
8. Nunmehr compilieren Sie das derart geänderte FULLSCREEN erneut und wenden es wie im Artikel beschrieben an. Sie werden sehen: kein Rand verschwindet mehr, denn es ist jetzt alles auf Ihre persönlichen Bedürfnisse (und natürlich auch Bildgrößen) eingestellt.

(Karl Tschentscher, Neustadt)







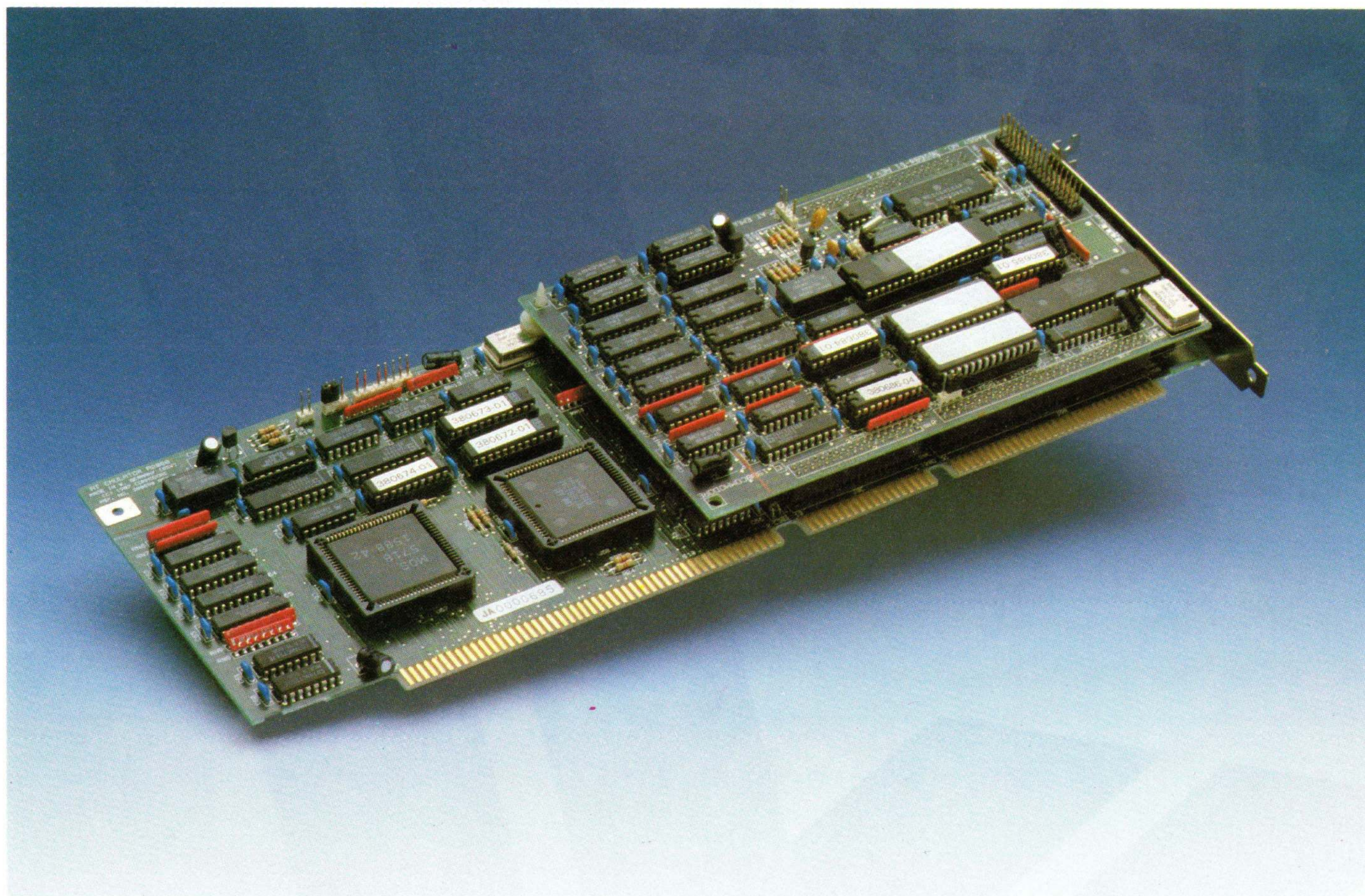


'MERLIN' COMPUTER GmbH

VON CHRISTIAN KELLER

Die AMIGA AT-Karte

DEM GLÜCKLICHEN SCHLÄGT KEINE STUNDE



Daß Commodore mit seinen Umsätzen zufrieden ist, dürfte hinreichend bekannt sein. Deswegen hat man auch wohl keine Eile bei der Fertigstellung der AT-Karte gehabt. Doch nun scheint es soweit zu sein.

AT steht bei IBM für "Advanced Technologie". Damit werden alle Rechner bezeichnet, die als CPU einen INTEL 80286 besitzen. AT ist aber auch die Abkürzung für das alte Testament, und die Bärte aller, die auf die AT-Karte gewartet haben, sind mindestens so lang wie die der im alten Testament erwähnten Personen. Nach der Erstankündigung im Frühjahr/Sommer 1987 folgte eine längere Phase von Ankündigungen, daß es endlich soweit sei, und daß die AT-Karte nun ausgeliefert würde. Doch auch 1988 brachte nicht den ersehnten Höhepunkt, 1989 war aber noch recht jung, als die endgültige Version die Labors von Commodore verließ. Auf den Test bislang verfügbarer Vorab-AT-Karten haben wir, nicht zuletzt im Interesse unserer Leser, verzichtet.

Da dieses freudige Ereignis sich erst am 5. Januar abspielte, war ein ausführlicher Test von Hard- und Software nicht mehr möglich. Daher wird dieser Bericht sich nur mit der AT-Karte selbst befassen. Im nächsten Heft wird dann die neue Treibersoftware besprochen, die sich auch für PC-Karte und Sidecar eignen soll. Die Software scheint zudem noch in Bewegung zu sein, da zu den Systemdisketten von Anfang Dezember noch ein zweiter Satz neueren Datums beigelegt war. Ein weiterer Anhaltspunkt für kleine Änderungen, die bis zur Auslieferung noch zu erwarten sind, ist die Tatsache, daß alles Beiwerk noch in Englisch abgefaßt ist.

Huckepack

Die AT-Karte nimmt sich im Vergleich zur PC-Karte recht gewaltig aus. Auf der Hauptplatine, die einen Brücken-Slot ganz in Beschlag nimmt, thront eine kleinere Huckepackplatine, die den RAM-Speicher, die BIOS-EPROMS und den Floppy-Controller beherbergt. Auf dem "Motherboard" findet man alle restlichen Bauteile wie die CPU, Sockel für den Coprozessor 80287, die hochintegrierten Customchips, welche diverse PC-Bauteile ersetzen, und alle Interface-

KiCK ME, AMIGA

R.C.T.

KiCK

Requester und Menü Construction Tool für den AMIGA

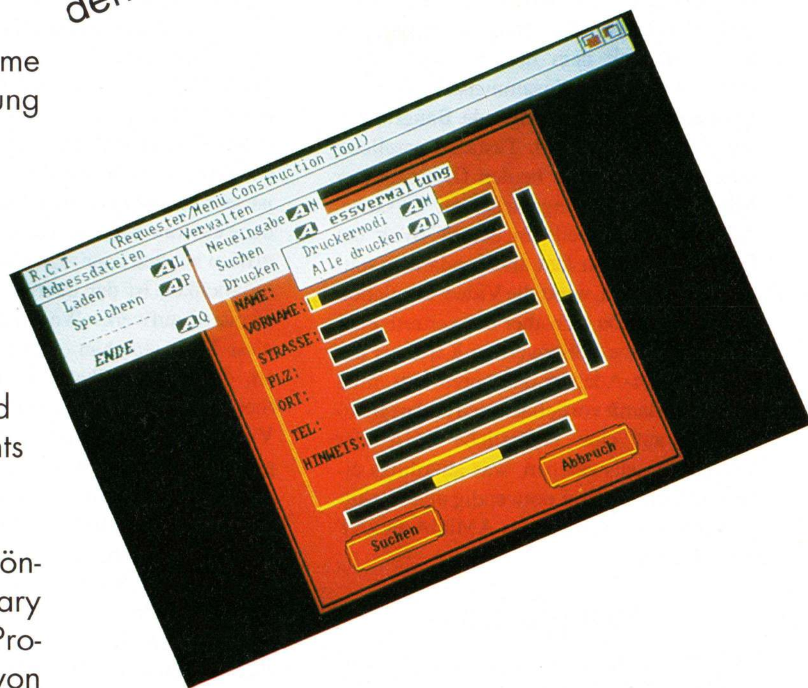
Der Weg in völlig neue Dimensionen

Wollten Sie nicht schon immer ihre Programme mit einer professionellen Benutzerführung versehen?

Mit dem R.C.T. ist das kein Problem, egal welche Programmiersprache Sie wählen.

Das Erstellen einer grafikorientierten Oberfläche ist für viele ein Buch mit sieben Siegeln. Das R.C.T. hilft Ihnen, Requester mit Gadgets und Images oder Menüs in verschiedenen Schriftfonts zu konstruieren.

Die konstruierten Requester und Menüleisten können mit den Funktionen der mitgelieferten Library verwaltet werden. Die Library kann von allen Programmiersprachen genutzt werden, auch von Basic aus.
Bei Programmiersprachen wie C oder Assembler besteht die Möglichkeit, die Funktionen der Library direkt an das Programm zu linken.
Der Sourcecode-Generator ermöglicht die Erzeugung von C-, Assembler- und GFA-BASIC-Source von den konstruierten Requestern und Menüleisten aus.



R.C.T. DM 129.-



Bestellcoupon

Bitte senden Sie mir:

☐ R.C.T. DM 129.-

Name

Straße

Ort

Datum Unterschrift

zuzüglich Versandkosten:
Inland DM 7.50
Ausland DM 10.-

Auslandsbestellungen nur gegen Vorauskasse
Bei Nachnahme zuzüglich DM 3.50 Nachnahmegebühr

MAXON-Computer / Industriestr. 26 / 6236 Eschborn

Bauteile (DualPortedRam, Adreß- und Datenbuslogikbausteine), die man in ihren grundlegenden Eigenschaften von der PC-Karte ja schon kennt. Die Taktfrequenz der AT-Karte beträgt 8 MHz. Da nicht erwartet werden muß, daß Commodore hier irgendwelche Böcke geschossen hat (wie zum Beispiel Speicherzugriffe mit diversen Wait-States, Stand der Technik ist hier 12-16 MHz ohne Waits), erübrigen sich hier die vielbeschworenen Benchmarks. Ein 8 MHz-AT ist nun mal etwa 4mal so schnell wie ein 4,77 MHz-PC, da ändert auch kein krampfhaftes Messen etwas daran. Man sollte sich übrigens nicht von den extraorbitant hohen Werten des Norton-SI beeindrucken lassen, da dieses ansonst feine Programm hier leider einige systematische Fehler begeht. Die Erweiterung mit einem numerischen Coprozessor ist vorgesehen, zur Installation muß aber die Zusatzplatine abgenommen werden.

Grafik

Da sich hier wenig Besonderheiten finden lassen, wird jetzt erst einmal auf die AMIGA-bedingten Besonderheiten eingegangen. Der Leistungsumfang der Emulation deckt sich mit dem der PC-Karte. Vom AMIGA werden in Zusammenarbeit mit der Emulationshardware folgende Baugruppen zur Verfügung gestellt: Tastatur, paralleler Druckerport und die beiden Grafikadapter MDA und CGA. Angesichts der Tatsache, daß sich mittlerweile in der PC-Welt die EGA-Karte schon fast als Standardausstattung etabliert hat, wirkt die Video-Emulation recht dürftig. Da sich aber mit vertretbarem Aufwand keine höherwertige Karte als der ColorGrafics-Adapter emulieren läßt, muß man wohl damit vorliebnehmen und gegebenenfalls auf eine zusätzliche EGA-oder VGA-Karte ausweichen, was allerdings einen zweiten Monitor notwendig und so einige Vorteile des Gespannes AMIGA-PC/AT zunichte macht.

Installation

Die Installation der AT-Karte ist recht problemlos. Die Karte braucht nur in einen der beiden Brückensteckplätze eingefügt und mit der Floppy verbunden zu werden. Zusätzlich muß man noch eine mitgelieferte Lithiumbatterie an einen Stecker auf der Huckepackplatine anklemmen und die Batterie im Gehäuse unterbringen. Hat man noch andere Steckkarten im AMIGA, wird dies zum Problem, da das Anschlußkabel recht kurz ist. Das selbige gilt für das Floppy-Kabel, auch dieses stört etwas, wenn weitere Karten installiert sind.

RAM

Die AT-Karte ist mit insgesamt 1024 kByte RAM-Speicher ausgestattet. Dieser teilt sich standardmäßig in 640 kByte RAM für normale DOS-Applikationen und 384 kByte Extended-Memory auf. In dem 640 kByte großen Bereich kann jedes Programm gestartet werden, der Extended-Speicher kann nur zu besonderen Zwecken benutzt werden. Als Hauptanwendung ist die RAM-Floppy anzusehen, die in diesen Bereich mittels Option in der Config.sys ausgelagert werden kann und so keinen wertvollen DOS-Speicher beansprucht. Ein anderer Verwendungszweck für diesen RAM-Bereich ist die Auslagerung von Datenpuffern, wie es zum Beispiel Turbo-Pascal oder Turbo-C unterstützen. Der komplette Speicher ist paritygeprüft.

Floppy

In der Grundkonfiguration ist die A2286 mit einem 5,25-Zoll-High-Density-Diskettenlaufwerk ausgestattet. Dieses Floppy-Laufwerk kann zwei Diskettenformate schreiben. Im HD-Modus, der als AT-Standard fungiert, wird eine Diskette mit 80 Spuren und 15 Sektoren pro Spur beschrieben und kann so 1,2 MByte Daten speichern. Hierzu sind allerdings die teuren HD-Disketten vonnöten. Zusätzlich kann auch das normale PC-Diskettenformat geschrieben werden, was den Datenaustausch mit PCs ermöglicht. Ein externes Laufwerk kann im Gegensatz zur PC-Karte nicht angeschlossen werden. Dies ist sehr bedauerlich, da so nicht auf vorhandene externe AMIGA-Laufwerke zurückgegriffen werden kann. Einzige Erweiterungsmöglichkeit ist der interne Anschluß eines zweiten Laufwerks. Wegen der Platzverhältnisse bietet sich hier ein 3,5-Zoll-Laufwerk an, das Anschlußkabel besitzt aber einen Stecker für ein zweites 5,25-Zoll-Laufwerk. Vielleicht ändert sich dies aber bis zur endgültigen Lieferversion. Das Bios unterstützt erfreulicherweise die Installation von 3,5-Zoll-Laufwerken in den 720 kByte- und 1,44 MByte-Formaten, was nicht selbstverständlich ist bei AT-Kompatiblen.

Der integrierte Floppykontroller kann desaktiviert werden, falls man einen AT-üblichen Kombi-Controller (Floppys und Harddisks) einbauen will. Es ist aber auch die Verwendung einer PC-Filecard oder anderer PC-Festplattenkombinationen möglich.

Setup

Jeder AT-Kompatible verfügt über einen eingebauten CMOS-Baustein, der neben einer Uhr, die batteriegepuffert ist, noch einige RAM-Zellen, in der die Systemkonfiguration abgespeichert wird, enthält. Hier werden Daten über installierte Disketten- und Harddisklaufwerke und deren Typ abgelegt, der beim Start gültige Videomodus und die Speicherkonfiguration eingestellt. Ebenfalls wird hier das Vorhandensein eines mathematischen Koprozessors angezeigt. Dieses RAM wird bei ATs üblicherweise mittels eines Setup-Programmes, das von Diskette geladen wird, abgeändert. Besser ausgestattete ATs, und hierzu zählt hier auch die AT-Karte, verfügen über ein ROM-residentes Setup. So kann man über einen "Affengriff" das Setup-Programm aktivieren, was immense Vorteile hat. Es kann ja einmal zu der Situation kommen, daß durch eine falsche CMOS-Configuration keine Diskette mehr gelesen werden kann. Dann ist ein Setup-Programm auf Diskette wenig nützlich. Als Betriebssystem wird MS-DOS 3.3 mitgeliefert. Es umfaßt inklusive der Janus-Utilities 3 Disketten. Hier findet man noch ein zweites Setup-Programm, das wesentlich komfortabler ist als das ROM-residente, da es nicht mit dem Speicherbedarf knausern muß wie dieses.

Fazit

Als Preis werden von Commodore 2600.- - 2700,- DM angegeben. Ein recht stolzer Preis, wenn man dies mit den Angeboten von Billigmarken-ATs vergleicht. Auch ist die Differenz zur PC-Karte doch recht gewaltig. Hier bleibt wirklich die Frage offen, ob man mit einem eigenständigen AT oder der PC-Karte nicht den besseren Kompromiß eingeht. Bei einem etwas günstigeren Preis und einer besseren Unterstützung der Kooperation beider Rechner (bessere Dokumentation von Services usw., bzw. fertige Lösungen für Aufgabenteilungen) kann die AT-Karte eine leistungsfähige und vielseitige Erweiterung darstellen, die ihrem Preis gerecht wird. Bleibt abzuwarten, wann die Auslieferung hierzulande wirklich beginnt. Die englische Software (unterstützt z.B. die deutsche Tastatur nicht korrekt) und die englische Dokumentation lassen noch eine kleine Wartezeit angebracht erscheinen, bis man die AT-Karte in den Läden finden wird. Die Hardware jedenfalls ist fertig, davon kann ausgegangen werden.

Vorschau

Wie eingangs schon erwähnt, wird sich die KICKSTART im nächsten Heft eingehend mit der neuen Software für die Brückenkarten beschäftigen. Diese bietet einige interessante Features, wie zum Beispiel Zeitübertragung, Mausemulation oder Booten des PCs von einer AMIGA-Festplatte. Inwieweit diese ordnungsgemäß arbeitet, soll dann geprüft werden. Eine kleine Enttäuschung sei schon vorweggenommen: Mit der Sidecar, die mit einem eigenen Installationsprogramm von der neuen Software bedient wird, gibt es erhebliche Schwierigkeiten mit den Janus-Partitionen.

ENDE

DAS GFA-BASIC 3.0 BUCH

Lieferbar!

KLAUS SCHNEIDER,
OLIVER STEINMEIER,
PETER FRITZEN

Als optimale Ergänzung zum Handbuch des neuen GFA-BASIC 3.0-Interpreters bietet sich dieses Buch an. In zwei Hauptteilen wird zunächst eine systematische Einführung in die Programmierung von BASIC unter Berücksichtigung der besonderen Fähigkeiten von GFA-BASIC auf dem Amiga gegeben. Hier werden dem Neuling vom ersten Einzeiler bis zu abstrakten Datentypen alle Möglichkeiten der strukturierten Programmierung mit zahlreichen, durch Flußdiagramme transparenter gemachten Beispielen nahegebracht. Doch auch BASIC-erfahrene Programmierer lernen hier die neuen Strukturen kennen, die sich doch sehr von denen anderer Dialekte unterscheiden.

Der zweite Teil baut auf dem ersten auf und vermittelt weitere Kenntnisse der Programmierung, die anhand von Programmen, die wiederum ausführlich beschrieben und erklärt sind. Hier seien ein leistungsfähiges Grafikprogramm sowie zahlreiche Beispiele zur Betriebssystemprogrammierung genannt.



Die Benutzung der verschiedenen Libraries und Intuition-Funktionen wird detailliert erklärt, so daß die Verwendung dieser Möglichkeiten in eigenen Programmen keine Schwierigkeiten bereitet.

Durch zahlreiche Anhänge – neben den Tabellen finden Sie auch ein sehr ausführliches Stichwortverzeichnis – wird das Buch optimal ergänzt und kann problemlos zum Nachschlagen von Details benutzt werden.

AUS DEM INHALT:

Erklärung der Schleifen- und Programmstrukturen
- FOR-NEXT, WHILE-WEND, REPEAT-UNTIL, DO-LOOP
- Prozeduren, Funktionsunterprogramme und Verzweigungen
- Rekursion
- Beispielprogramme

Variablentypen und Arrays
- numerische und Zeichenkettenvariablen
- Arrays zur Aufnahme großer Datenmengen

Multitasking in GFA-BASIC
- Reagieren auf Ereignisse
- Zeitabhängige Prozeduraufrufe

Programmentwicklung
- Programmplanung und -entwurf
- strukturierte Programmierung
- TOP-DOWN-Prinzip
- Fehlersuche
- Debugging-Möglichkeiten

Dateiverwaltung
- sequentielle Dateien
- Random-Access-Dateien
- Funktionen und Befehle zur Diskettenverwaltung

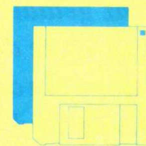
Grafikprogrammierung
- Die vielseitigen Möglichkeiten
- Turtlegrafik, Spriteprogrammierung
- Arbeiten mit mehreren Bildschirmen
- HAM-Modus
- Entwicklung eines vielseitigen Grafikprogramms

Sound- und Sprachprogrammierung
- Die Möglichkeiten der Befehle
- Verwendung der Anweisung in eigenen Programmen

Abstrakte Datentypen
- Verkettete Listen
- Bäume

Betriebssystem-programmierung
- Pulldownmenü-Steuerung
- Requester-Einsatz in Programmen
- Intuition- und Graphics-Library
- Aufruf von Systemroutinen

Für Einsteiger ★ Fortgeschrittene ★ und Profis



ÜBER 500 Seiten EINSCHLIESSLICH PROGRAMMDISKETTE

Dem Buch liegt eine Programm-Diskette bei mit über 150 Übungs- und Beispielprogrammen

BUCH & DISKETTE KOMPLETT
HARDCOVER DM 59,–

Priese sind unverbindlich empfohlene Verkaufspreise

Heim Verlag

Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

Schweiz: DataTrade AG
8021 Zürich
Langstr. 94

BESTELLCOUPON   AMIGA GFA 3.0 BUCH

Bitte senden Sie mir ________ St. DAS GFA-BASIC **3.0 BUCH** einschließlich Programm-Diskette für DM **59,–** zzgl. DM 5,– Versandkosten (unabhängig von bestellter Stückzahl)

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

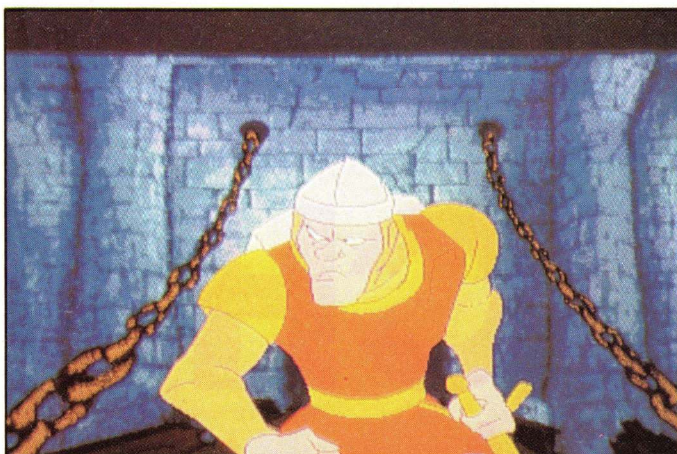
Name, Vorname ____________________

Straße, Hausnr. ____________________ PLZ/Ort ____________________

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte.

DRAGON'S LAIR

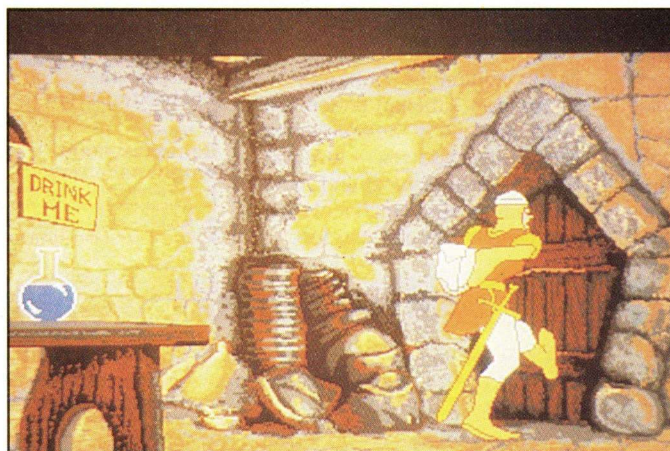
Rettet die Prinzessin



Der edle Ritter DIRK macht sich auf, die holde Prinzessin Daphne aus den Klauen des Drachen zu retten.

E**tliche Jahre ist es her, da machte DRAGON'S LAIR schon einmal Furore. Es nutzte zum ersten mal das Medium Laserdisk. Das Spiel war nicht ein Spiel im herkömmlichen Sinne. Die einzige Übereinstimmung bestand darin, daß die Hauptfigur, der edle Ritter DIRK, per Joystick gesteuert wurde, dies aber im sehr beschränktem Maße. Im richtigen Moment mußte der Joystick in eine bestimmte Richtung oder der Feuerknopf gedrückt werden. Daraufhin wurde von der Laserdisk eine neue Spielsequenz geladen. Alle Bewegungen wurden zeichentrickähnlich animiert. Direkte Einflüsse auf die Bewegungen hatte der Spieler nicht. Durch die Schnelligkeit der Laserdisk und die sehr gut gezeichneten Bilder entstand ein ansprechendes und unterhaltsames Spiel. Die AMIGA Version unterscheidet sich in der Spielhandlung und der Steuerung nur gering von seinem Vorbild. Auch hier kann der Ritter nur zu bestimmten Aktionen verleitet werden, wenn der Joystick zum richtigen Zeitpunkt betätigt wird. Die globale Animation übernimmt der AMIGA. Dies geschieht in einer Art und Weise, die bisher von keinem AMIGA-Spiel realisiert worden ist. Der Spieler glaubt sich mitten in einem Zeichentrickfilm, der Sound steht hier ins Nichts nach. Digitalisierte Geräusche, Musik und Sprache sind hörenswert und versetzen den Spieler in einen erwartungsvollen Zustand.

DRAGON'S LAIR versetzt den Spieler in das tiefe Mittelalter, wo böse Drachen, Magie und Zauberei an der Tagesordnung sind. Das Land schreit nach Helden und Heldentaten. Der edle Ritter DIRK gedenkt, wieder einmal eine solche zu vollbringen, und die edle Prinzessin Daphne aus den Klauen des teuflischen Drachens SINGE zu retten. In der Burg 'Drachennest' (engl. DRAGON'S LAIR) müssen



Überall lauern Gefahren, ob heimtückische Fallen, wilde Monster oder mit Vorsicht zu genießende Einladungen.

zahlreiche Abenteuer überstanden werden, bevor das Ziel erreicht ist. In dunklen Räumen stürzen sich plötzlich eine Horde Monster auf DIRK, oder es gilt Fallgruben und zusammenstürzende Brücken zu überwinden. Doch Helden vollbringen eben Heldentaten.

Auf insgesamt 6 Disketten sind 130 Mbyte Grafik und Animation untergebracht, die nacheinander eingelegt werden müssen. AMIGA-Anwender, die mehrere Laufwerke ihr eigen nennen können, sollten gleich alle Laufwerke mit Disketten 'füllen'. Es schränkt zumindest das Wechseln der Disketten ein wenig ein.



Nicht nur mit Schwert, sondern auch mit viel Geschick müssen manche Abenteuer überstanden werden.

Für jedes Abenteuer stehen drei Bildschirmleben zur Verfügung. Im Gegensatz zu anderen Spielen führt ein Fehler unweigerlich zum Tod. Nur die richtige Bewegung des Joysticks zum richtigen Zeitpunkt läßt den Helden weiterkommen und das Punktekonto wachsen. Leider hat diese Realisierung der Steuerung einen gewaltigen Nachteil. Der Spieler gelangt oft nur durch Ausprobieren weiter. Manche Sequenzen lassen auch keinen näheren Schluß über die richtige Joystick–aktion zu. 'Probieren geht über Studieren' heißt die Devise. Für manchen AMIGA-Anwender ist vielleicht die ausgezeichnete Grafik und Animation Entschädigung, andere wiederum werden keinen Gefallen an dauerndem Probieren und Raten haben. Die mitunter langen Ladezeiten der einzelnen Sequenzen sind dem Spielfluß auch nicht gerade dienlich.
Da DRAGON'S LAIR einen Kopierschutz besitzt, kann das Spiel auch auf keine handelsübliche Festplatte kopiert werden, obwohl laut Hersteller die Zusammenarbeit nur mit einer Comspec-Festplatte mit SCSI-Controller möglich ist. Unsere Versuche, das Spiel auf eine andere Festplatte zu kopieren, verliefen leider im Sande.
Einige Optionen haben die Programmierer dem Spiel zusätzlich mitgegeben. Die Auflösung kann zwischen Interlace und Hires gewählt werden. Des weiteren kann der Low Pass-Filter aus- bzw. angeschaltet werden. Die Soundqualität wird dadurch gesteigert (nur A500 und A2000B).

(AK)

Anbieter: GTI, Oberursel
Preis: DM 129,.



TIPS

Für den Einstieg möchte ich an dieser Stelle noch ein paar Tips geben: In der ersten Sequenz stürzt der Held durch ein Loch in der Hängebrücke. Hat das Bild gewechselt, drücken Sie den Feuerkopf etwa 1 Sekunde, daraufhin zieht DIRK sein Schwert und holt zu einem Rundumschlag aus. Drücken Sie jetzt den Joystick nach oben. Gewand klettert der Held nach oben und rennt in die Burg. Bei der zweiten Sequenz sollten Sie die nette Einladung auf keinen Fall annehmen und in dem Moment, wo die Tür aufblinkt, den Joystick nach rechts drücken. DIRK besinnt sich eines Besseren und tritt, statt den angebotenen Trank zu genießen, die Tür ein. Danach wiederholt sich die erste Sequenz, allerdings ist der Blickwinkel jetzt seitenverkehrt. Die Vorgehensweise ist hingegen genauso wie oben beschrieben. Die nächste Sequenz ist zufallsbedingt, entweder man findet sich in einem Gewölbe, wo Fledermäuse ihr Dasein fristen, oder der edle Ritter findet sich in einem Vorraum der Burg wieder. Im ersten Fall müssen Sie den zusammenbrechenden Weg durch Drücken des Joysticks nach rechts bzw. links überspringen und im zweiten Fall den Feuerknopf drücken, nachdem der grüne Tentakel erschienen ist. Weitere Tips nimmt die Redaktion gerne entgegen.

SPITTING IMAGE

Wenn sich Prominente prügeln

In England kennt sie jeder - die witzigen Spitting Image-Puppen, deren Köpfe Karikaturen bekannter Persönlichkeiten darstellen: Ronny Reagan, Gorbatschow, Maggie Thatcher, Prinzessin Diana, Charles und viele andere werden in dieser Serie aufs Köstlichste veräppelt. Klar, daß zu einer so populären Fernsehserie (in Deutschland läuft sie übrigens in "3-Sat") ein Computerspiel nicht fehlen darf.

In Domarks neuem Spitting Image-Spiel steht ein erneuter Weltkrieg bevor. Aus diesem schrecklichen Krieg wird nach Auffassung des Herstellers ein Führer hervorgehen, der die Weltherrschaft an sich reißt. Mögliche Weltherrscher wären Gorbatschow, Reagan, Maggie Thatcher, Khomeini, der Papst und Präsident Botha. Damit dieses schreckliche Unglück nicht über die Menschheit hereinbricht, muß ein geschickter Spieler antreten. Auf dem Bildschirm sieht man die Weltkugel und die sechs Portraitköpfe der möglichen Herrscher. Mit dem Joystick wählen Sie beispielsweise Reagan aus. Danach eine zweite Figur, zum Beispiel den Papst. Das Szenario wechselt, und die beiden Prominenten stehen sich jetzt vor einem lustigen Hintergrund gegenüber. Im Falle Ronny Reagans sieht man im Hintergrund ein typisches amerikanisches Fast Food-Restaurant. Ronny und der Papst müssen jetzt gegenseitig aufeinander eindreschen, bis einer von beiden keine Lebensenergie mehr hat. Ronny versucht seine Kontrahenten stets mit einer Clownnase zu pieksen, während der Papst mit einem Banjo um sich haut. Der zuerst Gewählte, also in diesem Falle Ronny, wird vom Computer gesteuert und muß unbedingt besiegt werden, damit das Spiel weitergeht. Liegt Ronny besiegt am Boden, zeigt der AMIGA wieder die Weltkarte, und man sucht sich den nächsten potentiellen Weltherrscher aus, der dann von der an zweiter Stelle gewählten Persönlichkeit besiegt werden muß. Das geht solange, bis man alle fünf Herrscher vermöbelt hat, dann soll eine Überraschung passieren. Domark wollte in der Anleitung nicht verraten, was dann geschieht. Ich kann es Ihnen verraten: nichts Besonderes. Das Bild fängt an zu zittern, und es erscheinen Farbstreifen auf dem Bildschirm. Das soll wohl bedeuten, daß die Erde durch den Krieg zerstört wurde. Gewinnt der Spieler alle Duelle, ist er nicht Retter der Welt, sondern seine Spielfigur wird zum Herrscher. Logisch, die anderen potentiellen Herrscher hat er ja zuvor alle besiegt. Glauben Sie ja nicht, daß ich wochenlang "Spitting Image" gespielt habe, um das herauszufinden. Die fünf Prominenten zu besiegen, war ein Kinderspiel und hat nicht länger als 20 Minuten gedauert. Ich muß sagen, ich war von "Spitting Image" ein wenig enttäuscht. Das Spielziel ist viel zu leicht zu erreichen. Es ist ja ganz lustig , zu sehen, wie Gorbie in ausgelatschten Jeans versucht, mit Kung Fu seine Gegner fertigzumachen. Maggie boxt und Khomeini schlägt mit seinen Kasperpuppen um sich. Die Grafik ist wirklich gut. Über den Sound kann ich ebenfalls nicht klagen. Es geschieht jedoch zu wenig Neues. Da kann auch eine Zwei-Spieler-Option nichts retten. Schade um die gute Idee und die mitunter bösartigen Gags, die aber mit Sicherheit auch nicht jedermanns Sache sind. *CBO*



Die englische Serie Spitting Image ist bei Prominenten eher verpönt und gefürchtet. Jetzt lehrt der AMIGA auch Sie das Fürchten.

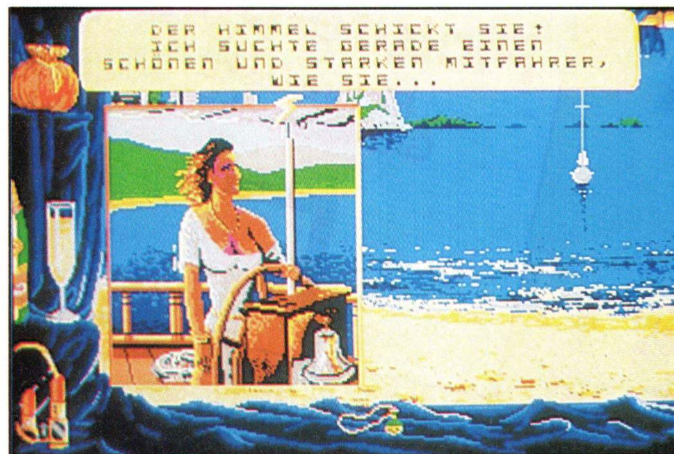
Info: Bomico



EMMANUELLE

Erotik am Computer?

Emmanuelle ist die Gattin eines reichen Franzosen, der ständig Geschäftsreisen unternimmt und deshalb nie Zeit für seine reizende Gattin hat. Zum Zeitvertreib legt sich Emmanuelle Marc zu, einen netten Hausfreund. Nach einigen Nächten voller Leidenschaft bekommt Emmanuelle Gewissensbisse, da sie immer noch ihren Ehemann liebt, und flieht vor der Wirklichkeit nach Rio. Marc reist ebenfalls nach Brasilien, um dort Emmanuelle zu suchen. Auf der Suche nach der bildschönen Frau sollte Marc erotische Erfahrungen sammeln, um Emmanuelle besser verwöhnen zu können. Während des Spielverlaufes kommt Marc mit anderen Frauen ins Gespräch, die man verführen kann. Einfühlsamkeit und Geduld sind erforderlich, ansonsten bekommt man einen Korb, oder es kommt zu einer Prügelei mit einem eifersüchtigen Ehemann oder anderen, die Streit suchen. Die Intensität der erotischen Erfahrungen wird in Form eines Lippenstiftes am linken Bildschirmrand gezeigt.



Emmanuelle verzückt auch auf dem AMIGA mit erotischen Reizen den unbedarften AMIGA-Anwender.

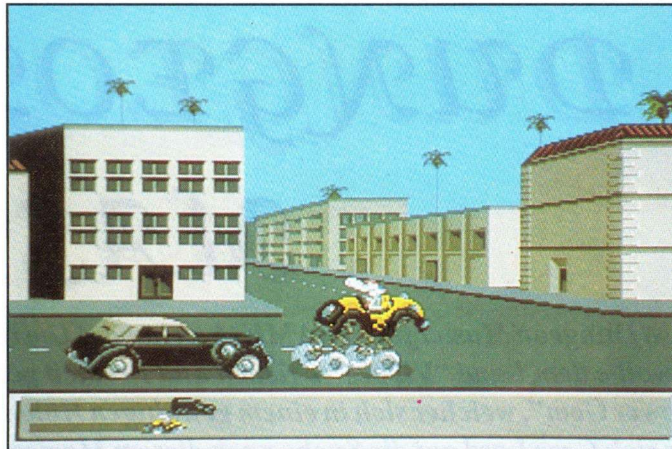
Das "Erotikbarometer" steigt umso höher, je öfter man Frauen an besonderen Orten verführt, und wenn man Abenteuer mit besonders schönen Frauen erlebt. "Emmanuelle" von "Coktel Vision" ist ein Adventurespiel mit einigen schwach animierten Szenen. Gespielt wird mit der Maus. Der Computer zeigt ein Bild, das Personen oder Gegenstände enthält. Bewegt man den Mauszeiger auf Gegenstände, kann man sie benutzen. Klickt man Personen an, kann man sich mit ihnen unterhalten. Die Charaktere geben dann ein paar Sprechblasen aus. Der Spieler hat die Möglichkeit, aus verschiedenen Sprechblasen eine Antwort herauszusuchen. Je nachdem, welche Antwort er gibt, reagiert die Person. Neben dem "Erotikbarometer" ist noch eine weitere Anzeige von Bedeutung. Ein kleines Säckchen, ebenfalls am rechten Bildschirmrand plaziert, gibt an, wieviel Geld Marc noch zur Verfügung steht. Der Ankauf von Informationen, Flugtikkets, Sekt oder ähnlichem kostet nämlich auch in Brasilien Bares. Wer viel reist, braucht viel Geld. Überall warten hübsche Mädchen darauf, verführt zu werden. Trotz des erotischen Themas ist "Emmanuelle" keineswegs pornografisch. Wenn es "zur Sache geht", sieht man ein Fragezeichen auf dem Bildschirm und eine Jalousie, hinter der mit Schatten zart angedeutet ist, was sich bei erotischer Tuchfühlung abspielen könnte. Grafisch gehört "Emmanuelle" zum gehobenen Standard. Bei der Animation müssen leider Abstriche gemacht werden, zu ruckartig sind die Bewegungen der Spielfiguren.

Ebenfalls negtiv fiel auf, daß die Rätsel, die das Spiel aufgibt, zu offensichtlich sind. Daher wird es nicht sehr lange dauern, bis man das Spiel gelöst hat. Ein großer Schwachpunkt liegt auch bei den Soundeffekten. Eine kurze, digitalisierte Titelmelodie und ein paar Geräuscheffekte sind zu wenig. CBO

Hersteller: Coktel Vision



***G**leichnamiger Film hat nicht nur wegen der aufwendigen Produktion für volle Kinokassen gesorgt, sondern auch wegen des putzigen Hasens Roger Rabbit, der Abertausende in seinen Bann gezogen hat. Die eigenwillige Realisierung des Films machte schon vor der Premiere in den Kinos Schlagzeilen, wurde doch die reale Welt mit der Zeichentrickwelt derart aufwendig gemixt, daß der Betrachter fasziniert auf die Leinwand starrte und meinte, daß es ganz normal wäre, daß ein Mensch aus Fleisch und Blut sich von einem TOON umarmen läßt. Ab sofort treibt der lustige Hase Roger Rabbit auch sein Unwesen auf dem AMIGA.*



ROGER RABBIT auf dem Weg zum INK & Paint Club.

WHO FRAMED ROGER RABBIT

Erinnern wir uns an das zigarrenrauchende Baby. Nach einer langen Ladezeit teilt es unserem Roger mit, daß der böse Richter DOOM und seine gewieften Wiesel TOONTOWN zerstören möchten. Schriftliche Informationen soll es im INK & PAINT CLUB geben, doch auf welcher Serviette? Nach diesem Vorspann, der übrigens hervorragend gezeichnet ist, kommt die erste Actionsequenz. Roger Rabbit sitzt am Steuer des TOON-Taxis und muß in einem Wettrennen seine Widersacher, die Wiesel, abhängen, um vor ihnen im INK & PAINT CLUB zu sein. "Nichts leichter als das", denkt man. Aber weit gefehlt: entgegenkommende Autos, Busse und Straßenbahnen machen es dem kleinen Helden nicht leicht. Zu allem Übel ist die Straße noch mit der TOON-vernichtenden Säure übergossen. Dank der vielen Möglichkeiten des TOON-Taxis kann sie aber übersprungen werden.

Nach abgeschlossener Fahrt muß Roger im CLUB von allen Tischen die Servietten abräumen. Leider erwischt er ab und zu ein Whisky-Glas, und wer den Film gesehen hat, weiß, daß Alkohol eine merkwürdige Reaktion auf Roger zeigt. Des weiteren sehen es die fleißigen Pinguine nicht gern, daß Roger die Tische abräumt, und decken sie neu ein. In die Hände des grimmig blickenden TOON-Gorillas sollte der kleine Roger ebenfalls nicht fallen, ansonsten findet er sich im Hinterhof zwischen Mülltonnen wieder. Findet er die Mitteilung, geht es weiter. Natürlich spielt seine entzückende und sehr sexy aussehende Frau auch mit. Sie wird in der letzten Aktionsequenz wichtig. Schauplatz ist die GAG FACTORY.

Roger Rabbit glänzt durch gute Grafiken, die sehr sehenswert sind, jedoch artet die Warterei auf das nächste Bild oder die nächste Actionsequenz zu längeren Kaffeepausen aus. Besonders, wenn der Spieler die Bilder schon kennt, kann sich das als störend erweisen. Die Actioneinlagen des Spiels sind sauber programmiert, könnten aber mitunter ein wenig besser umgesetzt werden. Beispielsweise ist das Scrolling bei der ersten Sequenz nicht gerade das Gelbe vom Ei. Der Sound ist durchschnittlich, paßt sich dem Spiel aber gut an. Parallelen zum Film sind deutlich zu erkennen, die Handlung stimmt aber nur im Groben mit der des Films überein. Roger Rabbit-Fans werden an diesem Spiel wohl nicht vorbeikommen. AMIGA-Besitzer, die den kleinen Hasen nicht so ins Herz geschlossen haben, können aber auch Gefallen an dem Spiel finden. Trotz einiger kleiner Fehler kann zum Kauf geraten werden, viele lustige Datails machen die Fehler wett.

Vertrieb: BOMICO



Ein Herz und eine Seele, Roger mit seiner entzükkenden Frau.

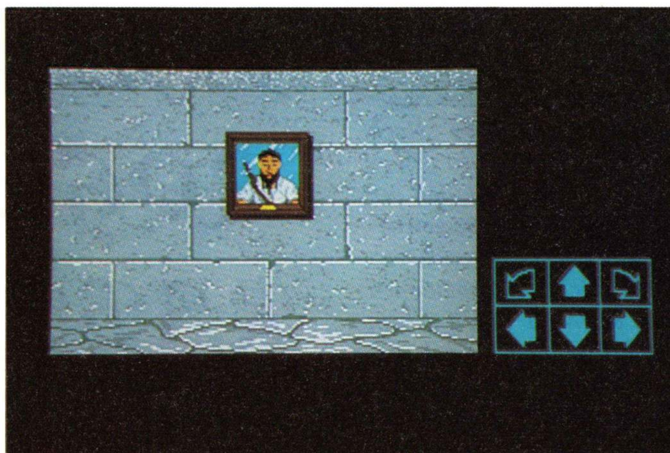




DUNGEON MASTER

Endlich ist Dungeon Master für den AMIGA zu haben! Die Hintergrundstory: Ein gewisser "Grey Lord" wollte dem Land "Viborg" Frieden und Freiheit geben. Dazu fehlte ihm allerdings noch der "Power Gem", welcher sich in einem gewaltigen Höhlensystem, dem Dungeon, befand. Also machte sich Grey Lord auf die Suche nach diesem Magischen EdelsteiNach einer gewissen Zeit hatte er den Edelstein auch gefunden, und er fing an, mit dessen Hilfe zu zaubern. Dabei erlebte er allerdings eine böse Überraschungen.

Der Zauberspruch des Edelsteins richtete sich plötzlich gegen ihn. Nach einer gewaltigen Explosion löste sich die böse Wesenshälfte von Grey Lord. Diese böse Hälfte nannte sich "Lord Chaos". Um seine Macht zu festigen, sucht Lord Chaos nach dem Power Gem, der nach der Explosion noch immer irgendwo im Dungeon liegen muß.

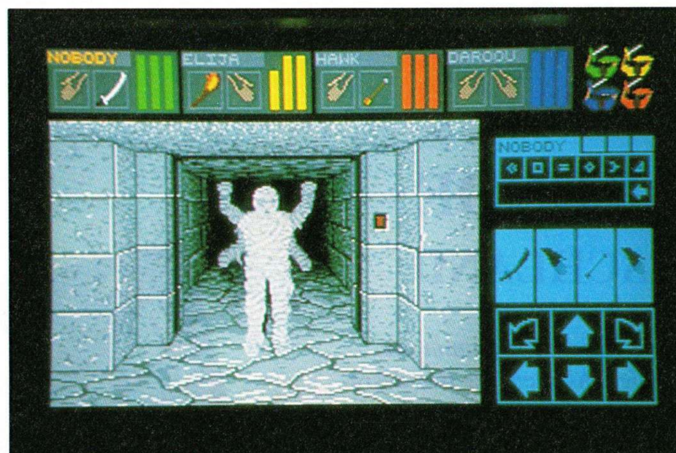


Mehrere Abenteurer haben schon von Grey Lord die Aufgabe erhalten, den Power Gem zu finden, aber noch hat es keiner geschafft. Um weitere Abenteurer gehörig abzuschrecken, hat Lord Chaos schon 24 davon in 24 Spiegeln eingeschlossen. Diese Spiegel befinden sich in der Halle der Champions. Sie können nun von den 24 Abenteurern in den Spiegeln vier Charaktere auswählen. Sehen Sie sich aber die Charaktere genau an. Hat man sich nämlich einmal festgelegt, kann man später nichts mehr daran ändern. Sobald vier Abenteurer gewählt wurden, sollten Sie diesen Spielstand abspeichern.

Das Besondere an Dungeon Master ist, daß Sie immer aus dem Blickpunkt eines Abenteurers schauen, und sämtliche Gegenstände werden nach jedem Schritt größer oder kleiner. Auch die Monster, die es auf die Gruppe der Abenteurer abgesehen haben, sind schon sehr früh auszumachen. In einem solchen Fall kann man sich erstmal verstecken und abwarten, was diese tun werden. Es gibt im Dungeon einige Puzzles und Rätsel zu lösen. Wenn man sie gelöst hat, dringt man immer tiefer in das Labyrinth ein. Die Kämpfe werden alle in "Echtzeit" ausgetragen. Das heißt, sollten Sie einige Sekunden während des Kampfes zu passiv sein, bekommen Sie gleich ein paar Hiebe zusätzlich. In Magie ausgebildete Charaktere können Zaubersprüche benutzen, womit Sie einige der folgenden Effekte erzielen können, z.B.: Türen öffnen, Gegner angreifen und Räume magisch ausleuch-



ten. Jeder Zauberspruch besteht aus mehreren Silben. Diese Silben werden durch grafische Symbole dargestellt. Die Schwierigkeit besteht allerdings darin, daß ein Zauberspruch aus mehreren Symbolen besteht, die man in der richtigen Kombination zusammensetzen muß. Im Dungeon kann man auch einige Schriftrollen finden, auf denen Zaubersprüche erklärt sind. Ein Magiekundiger, der in jeder Gruppe dabeisein sollte, müßte im Laufe des Spiels die Zaubersprüche lernen, um später eigene Sprüche zu entwickeln. Eine Beschreibung der Silben befindet sich im Handbuch des Spiels. Achten Sie während des Spiels auch ab und zu auf Ihren Nahrungsvorrat, damit Sie nicht verhungern oder verdursten. Es ist auch von Vorteil, das Spiel zwischendurch einige Male abzuspeichern, damit Ihnen nichts verlorengeht.



Dungeon Master ist in der Grafik sowie im Sound gut. Es ist ein herrliches Spiel für jeden Fantasiespieler. Hoffentlich kommen Fantasiespiele solcher Art noch öfter auf den Markt, was mit Sicherheit von jederman sehr begrüßt würde. Doch einige Übel hat daß Ganze noch. Man kann Dungeon Master nur spielen, wenn man mindestens 1 Megabyte hat. Auch ist es für den Englischlaien schwierig, mit dem leider englischen Handbuch zurechtzukommen.

H. Gumprecht





AMIGA... Hast du keinen - Hol dir einen!

SOFTWARE MADE IN GERMANY

ERDKUNDE
DEUTSCHLAND

Sie haben mit diesem Programm verschiedene Möglichkeiten, die Bundesrepublik Deutschland und die DDR kennenzulernen.

Städte, Länder, Flüsse, Gebirge

Städte: mit geschichtlicher Entwicklung, heutiger Industrie, bekannten Sehenswürdigkeiten.
Länder: geschichtlicher Hintergrund, industrielle Entwicklung.
Flüsse: von der Quelle bis zur Mündung mit den wichtigsten Nebenflüssen.
Gebirge: ihre Lage, Zugehörigkeit, höchste Erhebungen.
Quiz: zum Entspannen mit 250 verschiedenen Fragen.
Mit diesem Quiz kann der Benutzer das Erlernte unter spielerischem Aspekt testen und somit auch sein Allgemeinwissen prüfen.

Preis: DM 49.90



VIRUSKILLER
S.Y.S.

Zwei Programme die absolut wichtig sind:

1. Bootblock-Retter

Sichern Sie den Bootblock Ihrer wertvollen Disketten ab!

2. Virus-Checker

Überprüft Ihren Computer und alle Disketten auf Viren und elimeniert sie auf Wunsch!

Einmal benutzt, schon Geld gespart !!!

Preis: DM 49.90

ENGLISCH
KURS I

Dieses Programm bietet allen Amiga-Besitzern die faszinierende Möglichkeit, den Amiga als vollwertigen Lehrer/Lerncomputer einzusetzen. Auf spielerische Weise wird dem Anfänger von den einfachsten Zügen der englischen Sprache bis zu fortgeschrittenen Redewendungen alles beigebracht. Es beinhaltet Training des Vokabulars, Übungen zur Grammatik, Satzbau, Übersetzungen usw., in anderen Worten alles bis auf die englischen Aussprache.

Preis: DM 49.90

Weitere Lernprogramme (in Vorbereitung):

- Geschichte Amerika
- Erdkunde Amerika
- Mathematik (Kurvendiskussion)
- Geometrie
- Physik
- Englisch II

Erhältlich auch im gut sortierten Fachhandel und in Filialen von Allkauf, Hertie, Kaufhof und Neckermann
Fachhändler für Österreich und Schweiz:

INTERCOMP
Geschwend 163
A-6932 Lingen

MEGA SHOP
Falkenplatz 7
CH-3012 Bern

Bestellcoupon

Hiermit bestelle ich:

☐ Erdkunde Deutschland	DM 49.90
☐ Englisch Kurs I	DM 49.90
☐ Viruskiller S.Y.S.	DM 49.90

Versandkosten:

Inland	DM 2.50
Ausland	DM 5.00
(nur gegen Vorauskasse)	
Nachnahme zuzgl.	DM 3.50
(Nachnahmegebühr)	

Name: ____________________
Straße: ____________________
Ort: ____________________

CDC
Computer - Dienstleistungen

Louisenstraße 115
6380 Bad Homburg
Tel.: 06172- 2 47 48 / 20 7 99

Ziel des Spieles ist es, den Schwarzen Lord, der ein Helfer von Nikademus, einem bösen Zauberer, der das Land Gelnor mit Hilfe von Monsterarmeen unterjocht hat, ist, zu besiegen. Da man daran leicht verzweifeln kann, wollen wir Ihnen einige Tips zur Lösung der Aufgabe geben.



PHANTASIE I

Tips und Lösungshinweise

Man sollte darauf achten, daß eines der Gruppenmitglieder ein Minotaur ist, da man diesen später benötigt! In Gelnor gibt es in vielen Landschaften Herbergen, in denen die Gruppenmitglieder für Gold alle Wunden heilen, aber auch alle verlorenen Magiepunkte wiederbekommen können. In den Magic Pools kann man ein Attribut wie Stärke oder Konstitution steigern. Einer dieser Pools ist nur mit dem Zauber Transportation nach Phantasia zu erreichen. Der Ausgang von Phantasia liegt im Süden.

Im ersten Dungeon, nordwestlich von Pelnor, kann man oben rechts im Raum eine Truhe mit etwas besserer Ausrüstung finden. Links oben im Raum ist ein illusionärer Schatz, der verschwindet, wenn man ihn zu nehmen versucht. Statt dessen wird man angegriffen. Etwas weiter unten sind zwei Räume, in denen man Knöpfe drücken kann. Der linke öffnet das verschlossene Tor, hinter dem man den Bewahrer der Air Rune findet. Beim rechten ertönt ein Alarmsignal, und man wird wieder angegriffen. Weiter unten wird man von einem Wasserstrahl erfaßt und nach rechts gespült. An dieser Stelle kann man nach unten durch die Wand gehen und kommt in den Raum, wo die Ausrüstung von Kilmor steht. An der Wand steht: Nifle Scro.

In der Kathedrale sollte man den alten Mann in Janitors Raum ignorieren, da er dann eine Geheimtür im Süden des Raumes aufschließt. Diese führt zu einem natürlichen Höhlensystem, in dem man den Schlüssel für einen Raum (etwas links von der Mitte) findet, den man sonst nicht aufbekommt. Wem die Monster, die nach dieser Tür kommen, zu gefährlich sind, kann einen Geheimgang auf der rechten Seite benutzen, um in den Raum dahinter zu gelangen, in dem man Scroll 10 und 11 bekommt. Den Schlüssel für die andere abgeschlossene Tür bekommt man links oberhalb von ihr bei den Orks, wenn man sie nicht angreift. Dort liegt Scroll 20. Im Raum des Bewahrers der Earth Rune muß man die Box verbrennen! In der Küche muß man *Nifle Scro* zum Gnom sagen. Sein Name ist *Niscosnat*.

In der Höhle der Echsenwesen muß man sich im unteren rechten Raum entscheiden, ob man eine hübsche Frau oder einen alten Mann retten will, da sich der Raum sehr schnell mit Lava füllt. Die Frau schenkt der Gruppe als Dank einen Edelstein. Der Mann ist der Onkel von Lord Wood und kann nur noch *Straticon Uble* und *48* sagen, bevor er stirbt. Etwas links von der Mitte geht von oben ein Weg nach Süden, und eine Stimme warnt die Schwachen vor dem Weitergehen. Dort kommen nacheinander 2 Feuerelemente. Hat man diese besiegt, kommt man zum Hüter der Fire Rune. Unten links ist eine Stelle, an der die Höhle einstürzt. Wenn man dann im Raum rechts davon (durch Geheimgang erreichbar) die Tür nicht aufbekommt, ist man gefangen.

Woodville

Im Woodville Inn befindet sich Kilmor in Raum 1. Wenn man ihm den Namen des Gnoms (Niscosnat) sagt, verrät er der Gruppe, daß man die Quelle betreten kann, wenn man im Tempel 79 wählt. In der unteren Quelle findet man daraufhin Scroll 19. Im Raum Nr.3 der Inn wird man gefangengenommen und muß 500 Goldstücke bezahlen, um wieder freizukommen. Im Records Office findet man einen Hinweis auf Nr 45. Wählt man diese Nr. im Tempel, erscheint eine Tür im Norden der oberen Quelle. Geht man dort hinein, wird man am Ende des kurzen Ganges hinter die Taverne teleportiert. Dort findet man starke Gegner, aber am Ende wartet das God Knife, eine

Waffe mit dem Waffenwert 22! Im Scroll-Shop sollte man für 250 Goldstücke ein paar Scrolls kaufen, da man dann in den Storage-Raum kommt, von dem eine Geheimtür in den Secret Storage-Raum führt, in dem man Scroll 4, 5 und 14 findet. Als letztes wählt man im Tempel die 48 und kann nun Scroll 12 nehmen.

Lord Woods Castle

Vom linken Closet aus geht eine Geheimtür rechts zu einem kleinen Gang. Dort findet man Scroll 6 und 7. Im Closet steht *SWWSSESSWWWSSEEE-EENEEE*. Diese Himmelsrichtungsfolge beschreibt den sicheren Weg im unteren linken Raum. Alles andere sind Fallen. Am Ende gibt es etwas Gold als Belohnung. In einem Raum rechts (etwa in der Mitte) erhält man die Water Rune.

J.R.Trolkiens Castle

Die Gefängniszelle hat im Süden einen geheimen Ausgang, von dem wiederum ein geheimer Weg nach Westen abzweigt. Dieser führt in den Secret Control-Raum. Dort sind drei Hebel. Nr.1 öffnet die Gefängniszelle. Links von dieser ist ein Raum, in dem ein Schlüssel zu finden ist. Dieser öffnet die Ausgangstür des Schlosses. Hebel 2 öffnet den normalen Ausgang der Arena. Hebel 3 öffnet die Südtür von J.R. Trolkiens Raum. Dort sind die ersten drei Ringe zu finden.

Dungeon der Bleebs

Die blauen Pools verursachen Schaden. Mit den grünen Pools kann man sich teleportieren. Allerdings werden manche von ihnen erst aktiviert, wenn man vorher einen roten Pool betreten hat. Es gibt einen roten Pool, der die Richtung eines grünen Pools ändert. Dies muß man tun, um in den großen Raum zu kommen. Dort gibt es verschiedenfarbige Türen. Durch die rote Tür kommt man weiter. Im Raum danach drückt man den grünen Knopf. Die Antworten der drei Fragen sind: *Rot, Rot/Grün* und *Rot*. Damit hat man die nächsten 3 Ringe bekommen.

Tempel der Dosnebian Clerics

Man kann ihn nur betreten, wenn einer der Gruppenmitglieder ein Minotaur ist! Beim Tor gibt man *Reven Tonem* ein oder benutzt den Geheimgang über dieser Stelle! Nachdem man die schmale Statue des Minotaur bewegt hat, ist in der linken unteren Ecke des Haupttempels ein Geheimgang, der nach links zur Library führt. Von dieser führt ein Geheimgang nach Süden, von dem aus wiederum einer nach Osten zu 3 Gefängniszellen führt. In der untersten Zelle ist Lord Wood, der nach seiner Befreiung Scroll 16 übergibt. In der mittleren Zelle sitzt Kilmor, der einen Schlüssel überreicht, mit dem man den Raum aufschließen kann, hinter dem 2 weitere Ringe in einer Truhe liegen. Von diesem Raum aus führt ein geheimer Gang nach Norden zu einem zweiten Ausgang.

der Mitte ein Gang, durch den man zu Zeus persönlich kommt. Dieser gibt einem dann die God Rune, mit der man das Dungeon vom Schwarzen Lord betreten kann.

Der Schwarze Lord

Nachdem man schwere und schwerste Kämpfe überlebt hat, kann man irgendwann den Thronraum des Schwarzen Lords betreten, woraufhin dieser die Flucht ergreift. Im Thronraum zerstört man als erstes Gold, Edelsteine und den Thron, da der Schwarze Lord seine Kraft daraus schöpft. Danach macht man sich an die Verfolgung, die in einem Geheimgang über dem Thron beginnt. Am Ende drückt man die Option *Attack*, worauf er mit einem Zauberspruch die Felsen in eine Nord-Süd-Passage teilt. Nun geht man als erstes nach Norden,



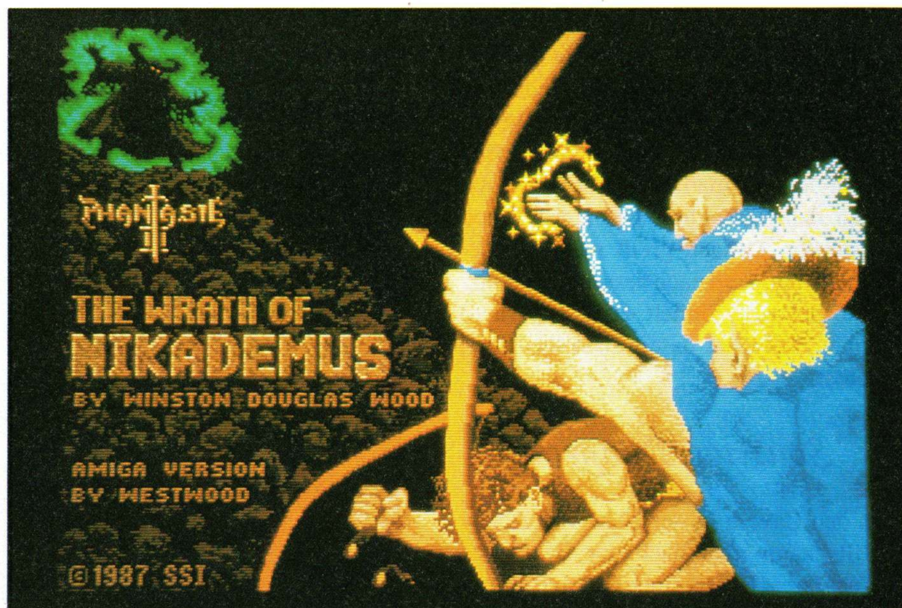
White Castle

Nachdem man sich mit dem Spruch Transportation zur Stadt Olympus teleportiert hat, kann man sich auf den Weg ins Weiße Schloß begeben, wo die Götter wohnen. Den River Styx kann man etwa in der Mitte überqueren. Im Schloß muß man 6 Prüfungen über sich ergehen lassen. Dazu muß man alle 4 Runen und die Ringe besitzen (ich hatte nur 6 Stück bei mir) und einen kleinen Kampf überleben. Danach öffnet sich im Süden in

um sich die Wand des Nikademus zu holen. Dort wird durch ein kleines Beben der Rückweg verschüttet, jedoch kann man mit *Wave Wand* die gesamte Gruppe in die Mitte des Dungeons teleportieren, wo man auf Filmon the Sage trifft und Scroll 17 findet. Filmon teleportiert die Gruppe auf Wunsch zum Ausgang oder zum Schwarzen Lord, um ihm den Garaus zu machen. Hat man dies geschafft, wird man zu Zeus teleportiert, erhält dort einen Zauberspruch und hat das Spiel gelöst!

Horst Gumbrecht

Das Ziel des Spiels ist es, Nikademus in der Netherworld aufzuspüren und zu vernichten.



PHANTASIE III

Tips und Lösungshinweise

Pentragon Town Archives

Im Süden von Pentragon liegt das erste Dungeon, in dem man wieder auf Filmon the Sage trifft, der der Gruppe von hier aus die Aufgaben stellt. Der 1. Auftrag ist, Kilmor aus der Hall of Giants zu retten. Betritt man diese, geht es gleich über eine Falltür in einen Raum, der eine Geheimtür im Norden hat, der angrenzende Raum wiederum eine im Westen. Geht man nun den langen Gang nach Süden, kann man 6 Türen öffnen. Knopf Nr.3 öffnet Zelle Nr.3, die einen geheimen Ausgang im Westen hat, jedoch kommt daraus ein starkes Monstrum. Will man diesem Kampf entgehen, öffnet man auch Nr.4, aus dem ein weiteres Monster herauskommt, das das 1. angreift. Nach kurzem Kampf haben sie sich dann gegenseitig getötet, und der Weg ist frei. Knopf Nr.5 öffnet die Zelle, in der Kilmor ist, allerdings ist er inzwischen verhungert und hat nur Scroll 1 hinterlassen. Mit diesem geht man zu Filmon, der sagt, daß man an der Beerdigungszeremonie in den Dwarven Burial Grounds teilnehmen soll.

Dwarven Burial Grounds

Dort erscheint Nikademus und metzelt etliche Leute nieder. Wieder zurück bei Filmon, soll man nun zu Chronos ins Chrystal Castle gehen, um über eine Maschine zur Light Plane und daraufhin zum Light Castle zu gelangen, um mit der Light Fairy zu sprechen. Das Chrystal Castle hat oben zwei spiralförmige Gänge. Am Ende des linken stellt man die Maschine ein, am Ende des rechten aktiviert man sie. Die linke untere Tür des ovalen Mittelraumes führt zu einem Raum, der im Osten eine Geheimtür hat, nach der man den Key of Light findet, den man zum Betreten des Light Castles benötigt. Um zur Plane of Light zu gelangen, stellt man an der Maschine C 2 + ein, aktiviert und geht aus dem Gebäude hinaus.

Light Castle

In diesem Gebäude soll man nicht kämpfen, es sei denn zur Verteidigung, wenn man angegriffen wird. Der sichere Weg zur Light Fairy ist: *WSNSWSWWW* (gilt für die Kreuzungen). Hat man mit der Light Fairy gesprochen, ist der nächste Auftrag, zur Dark Plane zu gehen, um dort mit Lord Wood zusammen Nikademus zu zerstören. Allerdings muß man sich vorher bei dem Zauberer in den Gnome Catacombs den 57. Zauber besorgt haben, mit dem man Lord Wood zu Hilfe rufen kann. Außerdem benötigt man für das Dark Castle den Dark Key. Diesen findet man im Süden der Wrestling Arena, die in den Dwarven Burial Grounds zu finden ist. Da man nun den Light Key besitzt, kann man diesen Raum betreten, um sich den Dark Key zu holen.

Dark Castle

Das Dark Castle ist das genaue Gegenstück zum Light Castle, also ist der sichere Weg ESNSESEEE. Zuvor muß man allerdings oben links in einem Raum die Wand von Nikademus besorgt haben. Am Ende des Weges macht man nun die Option Break Wand. Daraufhin wird man zu einem Ausgang teleportiert, der in die Netherworld führt.

Netherworld

Da man in der Netherworld viel durch Lava laufen muß, ist es angebracht, einen großen Vorrat an Magic Potions mitzunehmen, um die Leute mittels

Heilzaubern wieder heilen zu können. Am Ende des Weges liegt nun das Gebäude, in dem Nikademus zu finden ist. Dieses Gebäude hat sehr viele Geheimgänge. Nikademus ist etwa in der Mitte zu finden. Hat man ihn entdeckt, benutzt man den 57.Zauber, und Lord Wood erscheint. Danach greift man Nikademus an und wird nach dessen Besiegung von Zeus persönlich zum Superhero ernannt und hat damit auch das Spiel gelöst.

Sonstige Dungeons

In den Cavern in der Nähe von Rocky Hills kann man nach der Besiegung eines Great Beasts einen Gnom Bow mit dem Wert 13 finden. Außerdem bekommt man hier im Wizzards Lab den Zauber Nr.57, den man benötigt, um Lord Wood in der Netherworld zu Hilfe zu rufen.

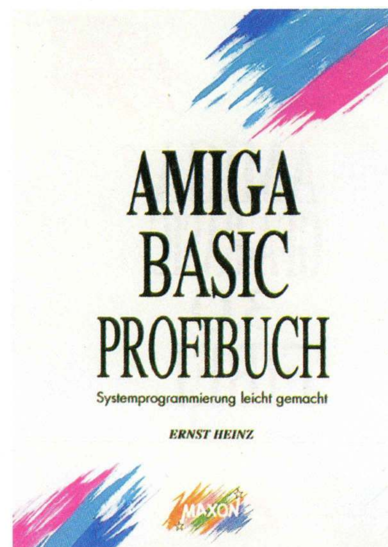


Im Zelt von Lord Wood kann man im Raum gegenüber von diesem eine Kiste mit einigen Sachen finden, unter anderem einen Bogen mit dem Wert 14.

In den Drachenhöhlen gibt es einige Drachenhorte, in denen ein paar nützliche Sachen zu finden sind.

Horst Gumbrecht

AMIGA BASIC Profibuch



NEU!

Für Theoretiker

Dieses Buch gibt Ihnen einen hervorragenden Überblick über die wichtigsten Möglichkeiten und Methoden der Systemprogrammierung unter AMIGA-BASIC. Neben einer detaillierten Beschreibung und Analyse der wichtigsten Funktionen finden Sie in diesem Buch eine ausführliche Erläuterung der verschiedensten System-Unterroutinen sowie den Aufbau des Gesamtsystems.

Für Praktiker

Im Hinblick auf die praktische Anwendung in eigenen Programmen dürfte die Sammlung von über 60(!) neuen AMIGA-BASIC-Unterroutinen zur Systemprogrammierung ein absoluter Lekkerbissen für jeden BASIC-Programmierer sein. Mit diesen neuen Unterroutinen werden die fantastischen Möglichkeiten des AMIGA nun endlich auch all denjenigen in einfacher Weise zugänglich gemacht, die selbst nicht tiefer ins Sytem eindringen möchten.

Für jeden

Blitzschnelle und variationsreiche Textausgabe, die Benutzung beliebiger Textfonts, die Erzeugung von Pull-Down-Menüs, die Gestaltung eigener Requester sowie die komplette Fenstersteuerung werden ab sofort für jeden BASIC-Programmierer leicht realisierbar.

Aber noch viel mehr Geheimnisse erwarten Sie während der Lektüre der einzelnen Kapitel.



Bestellcoupon MAXON Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Name: ______________________

Vorname: ______________________

Straße: ______________________

Ort: ______________________

Unterschrift: ______________________

Hiermit bestelle ich:

☐ Exemplar(e) von "AMIGA BASIC Profibuch". Mit Diskette für DM 59,00

☐ Vorauskasse

☐ Nachnahme

Versandkosten: DM 7,50
Nachnahme zuzgl. DM 3,50 Nachnahmegebühr.

Vertrieb in der Schweiz: DTZ DataTrade AG Langstrasse 94 Postfach 413 CH-8021 Zürich Tel.: 01/242 80 88 Fax.: 01/291 05 07
Vertrieb in Österreich: Dipl.-Ing. Reinhart Temmel Ges.m.b.H. & Co.KG Markt 109 A-5440 Golling Tel.: 06244/7081-17 Fax.: 06244/7188-3
Vertrtieb in Holland: JOTKA COMPUTING Postbus 8183 NL-6710 AD Ede Tel.: 08380/38731 Fax.: 08380/21675



Der Halb-2000er

2000er Tastatur am A1000

Die Tastaturen der neueren AMIGA-Modelle 500 und 2000 unterscheiden sich recht deutlich von der des Ur-AMIGA 1000. Wer seine A1000-Tastatur in den Knöpfehimmel geschickt hat oder auch nur ihrer überflüssig ist, kann seinen geliebten Rechner mit der eines A2000 (natürlich geht's auch mit der eines B2000) ausrüsten.

VON CHRISTIAN KELLER

Mit dem Erweitern der AMIGA-Produktlinie verband Commodore auch eine Erweiterung der Tastatur. Das Layout wurde etwas großzügiger arrangiert und der Tastenumfang der PC-Tastatur angepaßt. So fand man es sinnvoll, den Zehnerblock um die Tasten *NumLock*, *ScrollLock*, /, * und + zu erweitern.

Zwar haben diese zusätzlichen Tasten unter der Standardtastaturbelegung des AMIGA keine Funktion, sie werden aber von manchen Programmen (wie z.B. WordPerfect) benutzt. In absehbarer Zeit wird die Verwendung dieser Tasten sich ausbreiten, und es ist zu befürchten, daß nicht alle Programmierer die Weitsicht (oder besser Rücksicht) haben, an die A1000-Veteranen zu denken und die Funktionen optional durch Kombinationen von Tastendrükken zu emulieren. Wer dann nur eine "alte" 1000er-Tastatur besitzt, kann sich mit eingeschränkten Funktionen der Programme herumschlagen.

Es gibt aber noch andere Vorteile, die durch die "große" Tastatur entstehen. Sidecar-Besitzer können *AMIGA-N*, *AMIGA-P*, *AMIGA-S* usw. vergessen, wenn sie mit einer 2000er-Tastatur arbeiten. Ebenso ist es für jemanden, der auch viel am PC arbeitet, schwierig, sich an die doch recht beträchtlichen Unterschiede im Tastenabstand zwischen 1000er und PC zu gewöhnen. So mancher Fehlgriff ist vorprogrammiert.

Welcher Grund auch immer für einen Austausch spricht, das Vorhaben ist recht einfach durchzuführen, was den Umbau selbst betrifft.
Was zu Problemen führen kann, ist die Beschaffung eines passenden Steckers zum Tastaturanschluß des A1000. Dieser ist nämlich dem Ursprungsland des 1000er entsprechend ein hierzulande unüblicher schmaler amerikanischer Telefonstecker (Wie das verdammte Ding normgerecht heißt, weiß nur Herr Godot.).

Wer sein altes Tastaturkabel nicht verhackstückeln will, hat folgende Möglichkeiten:

- Man besorgt sich ein zweites Kabel im AMIGA-Ersatzteilhandel (Kleinanzeigen beachten, recht teuer mit ca. 35,- Talern).
- Man versucht es bei einem gut sortierten Telefonhändler (Nein, nicht unsere liebe Bundespost, die "nur für den Export"-Läden sind gemeint.).
- Der Onkel aus Amerika soll eins schicken (auf die schmalen Stecker hinweisen, sehr preiswert mit 2,- Dollar).
- Man baut den AMIGA auf die DIN-Buchse der Tastatur um. Dann entfällt der Adapter, respektive er ist ja schon im Rechner drin.

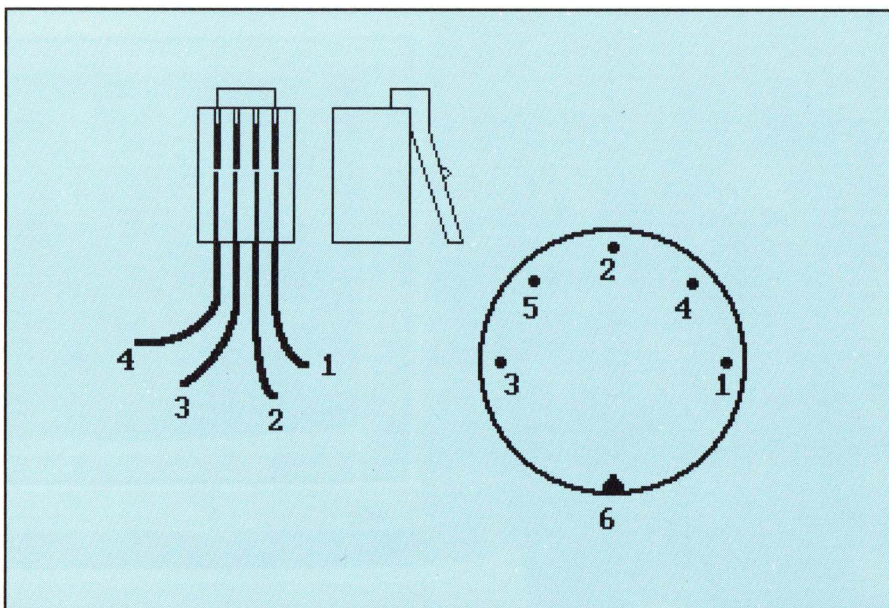
Was man neben dem Telefonstecker noch braucht, ist schnell aufgezählt:

1 5-polige DIN-Kupplung 180° mit Hülle

1 2000er-Tastatur

Als Arbeitsmittel benötigt man einen Lötkolben mit nicht zu großer Leistung.

Ist dieser Punkt zur Zufriedenheit geklärt, kann man seinen Freund zum Anschluß der neuer Tastatur vorbereiten. Man nimmt den Telefonstecker, an dem sich noch etwa 5 cm Kabel befinden sollten, und isoliert die einzelnen Adern etwas. Die Adern verzinnt man an den Enden und lötet sie an die entsprechenden Pins der DIN-Kupplung an. Die Verbindungsliste lautet wie folgt:



Telefonstecker	*Din-Buchse*
1	4
2	5
3	1
4	2

Die Numerierung der Pins finden Sie in obigem Bild. Die Pinnumerierung des Telefonsteckers ist in der Draufsicht gewählt. Die Nummern an der DIN-Buchse sind der Norm angepaßt. Das Bild zeigt einen Stecker von der Stiftseite bzw. eine Buchse von der Lötseite.
Hat man diese Verbindungen gelötet, ist die Sache auch schon fertig. Den Telefonstecker in den AMIGA und den Tastaturstecker in die Buchse einstecken und einschalten. Die Caps-Lock-Diode sollte kurz aufleuchten. Schon ist die Tastatur betriebsbereit.
Falls die LED der Tastatur nicht leuchtet oder ein anderes Verhalten an den Tag legt, sollten Sie den AMIGA ausschalten und Ihre Verdrahtung kontrollieren. So einfach wie sie ist, so leicht schleichen sich Fehler ein, wie ich selbst feststellen mußte. Der Rechner und die Tastatur nahmen die Falschverdrahtung zwar nicht übel, man sollte es aber nicht darauf anlegen. Seien Sie also trotz der simplen Verbindung nicht unaufmerksam.

Wenn Sie mangels Telefonstecker oder aus sonst einem Grund den direkten Anschluß der Buchse an den AMIGA vorziehen, gehen Sie am besten wie folgt vor:

- Bohren Sie ein Loch in die Rückwand des AMIGA.
- Nehmen Sie ein längeres Stück Kabel und stecken Sie es durch die Bohrung in der Rückwand.
- Bringen Sie auf der Innenseite eine Zugentlastung an. Diese kann aus einem aufklemmbaren Kunststoffring bestehen, oder, falls man diesen nicht erhält, notfalls auch aus einem Knoten im Kabel.
- Nun schließen Sie wieder die Kabel an die einzelnen Pins an.
- Praktisch ist, wenn man die Adern im Rechner direkt an die Anschlußdrähte der Telefonbuchse anlötet. So erspart man sich den Ausbau der Hauptplatine. Achtung, Pin 1 ist vom Geräteinneren aus betrachtet der links, Pin 4 der rechts außen.
- Beim Löten muß man Sorgfalt walten lassen, da die Abstände zwischen den Kontakten recht gering sind.

Auf diese Art und Weise haben Sie zwei Tastaturanschlüsse parallel. Sie dürfen aber auf keinen Fall zwei Tastaturen gleichzeitig anschließen.

Übrigens, von der 2000er-Tastatur gibt es verschiedene Ausgaben, die sich recht deutlich im Schreibgefühl unterscheiden. Da kommt es auf den Geschmack an. Funktionell sind die Tastenbretter alle gleich. Viel Spaß also mit Ihrem "Halb-2000er"

ENDE

VON CHRISTIAN KELLER

RAMBOOT

Booten von RAM-Disk - auch mit wenig Speicher

Mit der neuen Workbench 1.3 ist es möglich geworden, auch von einer RAM-Disk zu booten. Wie man dies bewerkstelligt, auch wenn man nicht gerade zwei oder mehr Megabyte besitzt, soll hier einmal gezeigt werden.

Das Hochfahren des Computers, zum Beispiel nach einem Absturz, ist aufgrund der langsamen Diskettenoperationen des AMIGA teilweise recht nervig. Wie schön wäre es da, diese Prozedur auf ein schnelleres Medium zu verlagern. Speziell die HardDisk-Besitzer ärgern sich darüber, daß sie ein so leistungsfähiges Speichermedium besitzen, aber immer noch von Diskette booten müssen. Durch die neue Kickstart 1.3 ist der Ausblick schon etwas rosiger geworden. Diese erlaubt ja grundsätzlich ein Starten des Systems auch von anderen Devices als von DF0:, dies aber nur, wenn die Geräte das auch selbst unterstützen. Wer sich jetzt einen bootfähigen Festplattencontroller anschafft, ist fein heraus. Neue Kickstart-Eproms eingesetzt, und schon kann man seinen 2000er ganz von Festplatte hochfahren. Was macht aber der Besitzer einer Festplatte, die dies nicht unterstützt? Er benutzt eine RAM-Disk.

Es handelt sich allerdings nicht um die normale RAM-Disk, die das Betriebssystem bisher auch schon zur Verfügung gestellt hat. Nein, was Neues muß her. Die bootfähige RAM-Disk heißt *Card*. *Card* wird bei der ersten (und nur bei dieser) Initialisierung gemountet. Der entsprechende Device-Treiber heißt ramdrive.device und befindet sich auf der Workbench 1.3 im *devs*-Ordner.

Der Mountlist- Eintrag sieht dann wie folgt aus:

```
CARD:
   Device           = ramdrive.device
   Unit             = 0
   Flags            = 0
   Surfaces         = 2
   BlocksPerTrack   = 11
   Reserved         = 2
   Interleave       = 0
   LowCyl           = 0 ; HighCyl = 21
   Buffers          = 5
   BufMemType       = 1
   Bootpri          = 120
#
```

An dieser Stelle soll nur auf die Punkte eingegangen werden, die sich von einem Eintrag unter DOS 1.2 unterscheiden bzw. neu hinzugekommen sind.

Bootpri - Dieser Eintrag gibt die Bootpriorität eines bootfähigen Devices an. Der erlaubte Wertebereich ist -127-128. -127 kennzeichnet, daß die Bootoption abgeschaltet ist.

BufMemType - gibt die Speicherart an, die zum Anlegen der Puffer benutzt wird. Dabei bedeutet:

```
0 , 1  Speichertyp beliebig
2 , 3  Chipmemory
4 , 5  Fastmemory
```

Zu beachten ist, daß der obige *Mountlist*-Eintrag nicht mit dem *Mount*-Befehl der Workbench 1.2 verwendet werden darf, da dieser den Eintrag *Bootpri* ablehnt. Der Befehl wurde unter DOS 1.3 daraufhin erweitert.

Startup-Sequence

Bei der Startup-Sequence bedarf es natürlich einiger Änderungen, die auf die veränderten Verhältnisse zugeschnitten sind. Wer wirklich zuviel Speicher hat, kann natürlich eine 80 Spuren-Card mounten und die Systemdiskette in der Startup-Sequence mittels Diskcopy einfach auf die *Card:* kopieren. Ich bin allerdings der Meinung, daß RAM für solche Zwecke zu kostbar ist und gehe daher einen anderen Weg (der allerdings eine Festplatte voraussetzt).

Beim ersten Starten des Systems von Diskette wird die Card: initialisiert und mit einem Minimalsystem eingerichtet. Danach übergibt die Startup-Sequence die Kontrolle an ein Batchprogramm auf der Festplatte. Bei einem Absturz oder Reset wird das System von der RAM-

Disk gestartet und die Festplatte gemountet. Danach wird die Kontrolle wieder an die Festplatte übergeben.

Inhalt der Card:

Auf der *Card:* muß zum Starten folgendes untergebracht sein:

1. einige CLI-Befehle, die zur Installation weiterer Devices und zu Wechseln auf diese benötigt werden. Dies wären zum Bespiel *Binddrivers*, *DJ-Mount*, bzw. *Mount*, *Execute*, *Assign*, *Wait* und *CD*.
2. der Devs-Ordner mit allen Device-Treibern, die benötigt werden, bevor man mit *Assign Devs:* der Festplatte zuweisen kann. Dies sind zum Beipiel *jdisk .device* oder der zu Ihrer Platte gehörige Treiber. Wird die Platte über *Mount* eingebunden, muß auch die Mountlist zu dem Gerät da sein.
3. Der L-Ordner sollte den Disk-Validator enthalten.
4. Libs muß die Janus.library enthalten.
5. Ein Verzeichnis *s:* mit einer Startup-Sequence ist natürlich auch notwendig.

Dies reicht, um den AMIGA zu starten. Mit einer AMIGA-seitigen Festplatte könnte die Startup-Sequence wie folgt aussehen:

```
Mount dh0:
Execute dh0:s/Startup-HD
Endcli >nil:
```

In der Batchdatei Startup-HD stehen dann alle weiteren Befehle. Als erstes sollten alle Verzeichnisse auf die Festplatte umassigned werden. Der Beginn des Batches sollte so aussehen:

```
assign    sys: DH0:
assign      t: sys : t
assign      c: sys : c
assign   devs: sys : devs
assign  fonts: sys : fonts
assign      l: sys : l
assign   libs: sys : libs
assign      s: sys : s
........
```

Hier folgen dann alle weiteren Befehle, die Sie sonst in Ihrer Startup-Sequence haben.

Besitzen Sie eine Janus-Festplatte, muß die Startup-Sequence auf der *Card:* wie folgt aussehen:

```
Binddrivers
Wait 40
DJMount
Execute jh0:s/Startup-HD
Endcli >nil:
```

Damit wäre alles geregelt, um nach einem Reset diskettenlos zu booten. Es fehlt nur noch die Startdiskette, mit der *Card:* installiert und für das Booten präpariert wird. Die Startup-Sequence entspricht derjenigen der *Card:*, ist aber noch um ein paar Zeilen erweitert. Nach dem *Mount-* bzw. *DJMount*-Befehl werden folgende Zeilen eingefügt:

```
Mount Card :
copy dh0 : cardinhalt to Card : all quiet
```

Auf Ihrer Festplatte haben Sie ein Verzeichnis *Cardinhalt*, das alle Dateien enthält, die oben angesprochen wurden. Das ist praktischer, als alle Dateien einzeln aus den verschiedenen Ordnern auf die *Card:* zu kopieren. Erstens erspart man sich das Erstellen der verschiedenen Unterverzeichnisse, und zweitens geht es auch schneller, weil der Copy-Befehl nur einmal von Diskette geladen werden muß. (Der AMIGA muß ja alle Befehle vor der Ausführung von Diskette laden.) Dritter Vorteil ist die bessere Übersichtlichkeit bei Änderungen.

Einen kleinen Schönheitsfehler hat diese Methode allerdings noch: Bootet man trotz *Card:* von Diskette, wird natürlich die *Card:* nochmal gemountet, was aber nur zu einer Meldung "Device already mounted" führt. Der Copy-Befehl wird aber unnötigerweise wiederholt, was Zeit kostet. Durch eine Änderung der obigen Sequence kann dem abgeholfen werden. Dabei gibt es zwei Möglichkeiten. Die erste läßt den Mountbefehl unberührt und verhindert nur das nochmalige Füllen der *Card:*:

```
Mount Card:
      IF NOT EXISTS Card:c/Binddrivers
copy dh0:card to card: all quiet
EndIF
Endcli >nil:
```

Mit dem Assign-Befehl der Workbench 1.3 geht es allerdings noch eleganter. Dieser bedient eine neue Option "list". Mit diesem Parameter gibt Assign einen Warn (Fehlerlevel 5) aus, falls man einen zu suchenden Volume-Namen angibt.

```
Assign  > nil:Card: list
If warn
        Mount Card:
        copy dh0:card to card: all quiet
Endif
```

Damit ist alles vorbereitet für ein schnelles Booten von RAM-Disk. Aber wie oben schon erwähnt, Kickstart 1.3 ist Grundvoraussetzung.

ENDE

VON ANDREAS KRÄMER

DER ASSIGN-BEFEHL

Der Schritt zur Ordnung

Befehle wie DIR, LIST oder TYPE werden noch leicht verstanden, anders verhält es sich mit Befehlen wie MOUNT oder ASSIGN. Letzteren möchte ich Ihnen ein wenig näher erklären, kommen Sie früher oder später doch um die Benutzung nicht herum. Spätestens wenn Sie im Besitz einer Festplatte sind oder viel mit der RAM-Disk arbeiten, werden Sie den Befehl nicht außer acht lassen können.

Um Ihnen den ASSIGN-Befehl näher erklären zu können, muß ich ein wenig weiter ausholen. Im mitgelieferten AMIGADOS-Handbuch findet man unter der ASSIGN-Erklärung folgenden Satz, der den Befehl zwar treffend beschreibt, mit dem der unbedarfte AMIGA-Einsteiger aber reichlich wenig anfangen kann:
"*Weist ein angegebenes, logisches Gerät einem Verzeichnis des Dateisystems zu*". Was ist z.B. ein logisches Gerät, das, wie wir im Satz lesen können, einem Verzeichnis, oder auf Englisch Directory, zugewiesen wird?

AMIGA-Einsteiger haben nach Erwerb ihres AMIGA so ihre Probleme, besonders mit dem Command Line Interpreter, kurz CLI, treten Schwierigkeiten auf. Stellt der Befehlszeilen-Interpreter (deutsche Übersetzung) doch ein enorm komplexes Kommunikationsmedium mit dem Rechner dar, das nicht so mir nichts dir nichts beherrscht werden kann.

Logische Geräte

Als logische Geräte werden unter AMIGADOS alle physikalisch vorhandenen Geräte oder die Peripherie bezeichnet, die vom Benutzer angesprochen werden können. Doch nicht nur das! Es sind unter anderem alle angeschlossenen Diskettenlaufwerke, die unter folgenden Gerätenamen angesprochen werden können, sofern sie angeschlossen sind: *DF0:*, *DF1:*, *DF2:*, und *DF3:*. Weitere logische Geräte sind mit *PRT:* und *PAR:* deklariert. Wie Sie vielleicht schon in Erfahrung bringen konnten, dienen beide der Ansteuerung eines Druckers, somit wird mit diesen Gerätenamen der parallele Port des AMIGA angesprochen. *PRT:* benutzt hierbei den unter Preferences eingestellten Druckertreiber; *PAR:* hingegen umgeht den Treiber und gibt die zu diesem Gerät gesendeten Daten direkt aus. Dies ist die Regel, ich möchte aber noch erwähnen, daß das logische Gerät *PRT:* auch der seriellen Schnittstelle zugewiesen werden kann. Die Zuweisung erfolgt mit dem Programm Preferences, das sich auf jeder Workbenchdiskette findet. Um die serielle Schnittstelle anzusprechen, dient der Gerätename *SER:*. Der Bildschirm kann ebenfalls angesprochen werden, dafür stehen dem Anwender die Gerätenamen *CON:* und *RAW:* zur Verfügung. Werden die logischen Geräte angesprochen, wird ein Fenster geöffnet, auf dem die Ausgabe dann stattfindet. Ein weiteres logisches Gerät ist das merkwürdige namens *NIL:*. *NIL:* ist schnell und einleuchtend mit einem Satz erklärt: Es ist ein Gerät, das es eigentlich gar nicht gibt. Bildlich kann sich der Anwender das Gerät als einen Draht vorstellen, der in der Luft hängt, also ein offenes Ende besitzt. Werden Daten an dieses Gerät geschickt, 'fallen' sie sozusagen ins Nichts. Die Geräte, die ich bisher erklärt habe, sind in der Regel nach dem Booten einer normalen Workbench 1.2 angemeldet, dies sind:

DF0, DF1, DF2, DF3, PRT, PAR, SER, RAW, CON und RAM.

Die Diskettenlaufwerke müssen jedoch angeschlossen sein.
Andere Geräte müssen erst angesprochen bzw. 'gemountet' werden. Nicht anders verhält es sich mit Festplatten. Die Anmeldung geschieht mit Hilfe des MOUNT-Befehls. Erst danach ist dem Betriebssytem mitgeteilt, daß da noch ein weiteres Gerät ist. In der Regel gehören dazu auch andere RAM-Laufwerke, Eprombänke oder die unter der Workbench 1.3 benutzten Geräte *NEWCON: SPEAKER:* oder *PIPE:*.
Alle bisher erklärten logischen Geräte sind physikalisch gesehen vorhanden, ob dies jetzt ein Diskettenlaufwerk oder nur ein spezieller Chip ist. Anders verhält es sich mit anderen logischen Geräten, die physikalisch gesehen nicht vorhanden sind und nur als Verzeichnisse existieren. Jetzt kommt der ASSIGN-Befehl ins Spiel. Das Betriebssystem unternimmt beim Booten bereits etliche Zuweisungen auf Verzeichnisse. Diese haben folgende Namen und Bedeutungen:

SYS:
stellt immer das Basisverzeichnis der Systemdiskette dar. Bei einem Zugriff auf *SYS:* wird immer auf dieses Basisverzeichnis zugegriffen, in der Regel das oberste Laufwerk der Boot-Diskette.

C:
definiert die AMIGADOS-Befehlsbibliothek. In diesem Verzeichnis sind alle CLI-Befehle abgelegt. Das Betriebssystem greift immer auf dieses Verzeichnis zu, sobald im CLI ein Befehl oder ein Programm aufgerufen wird, wobei zuerst im momentan aktuellen Verzeichnis nach dem betreffenden Befehl oder Programm gesucht wird. Nur bei Mißerfolg wird dann im C-Verzeichnis gesucht.

L:
das Bibliotheksverzeichnis. Hier sucht das Betriebssystem nach Programmteilen, die sich meistens aus Platzgründen nicht im Arbeitsspeicher befinden.

S:
ist das Stapeldateiverarbeitungsverzeichnis. In ihm findet der Anwender die wichtige Datei STARTUP-SEQUENCE. Bei jedem Bootvorgang wird auf diese Datei zugegriffen, die dann abgearbeitet wird. Der Anwender kann durch Änderung der Datei das System seinen Wünschen automatisch anpassen.

LIBS:
ist die Bibliothek für Bibiothekseröffnungen. Hier stehen die Bibliotheken, die sich nicht fest im ROM befinden und bei Bedarf nachgeladen werden müssen.

DEVS:
ist das Verzeichnis für Geräteeröffnungen. In diesem Verzeichnis sind alle Gerätetreiber zu finden.

FONTS:
ist das Verzeichnis für ladbare Zeichensätze. Wenn ein Programm auf einen neuen Zeichensatz zurückgreift, wird dieser von diesem Verzeichnis aus geladen.

T:
ist das Verzeichnis für zeitweilige Arbeitsdateien. Viele Editoren legen eine Backup-Datei in diesem Verzeichnis ab, bei Fehlern kann aus diesem Verzeichnis die zuletzt geänderte Datei kopiert werden.

Diese Verzeichnisse sind bereits intern mit dem ASSIGN-Befehl zugewiesen worden und können selbstverständlich, wieder mit dem ASSIGN-Befehl, umgelenkt werden. Ähnlich geht es mit den Diskettenlaufwerken, oder besser den eingelegten Disketten, jedoch muß hier einiges beachtet werden. Bei Festplatten oder den anderen physikalischen Geräten ist der ASSIGN-Befehl nur in 'einer Richtung' anwendbar, dazu später mehr.

Ein weiterer Hinweis: Mit Hilfe des CD-Befehls (Change Directory) kann schnell und einfach auf diese Verzeichnisse gewechselt werden.

```
CD C:
```

wechselt zum Verzeichnis C:, das als AMIGADOS-Befehlsbibliothek deklariert ist. Mit dem CD-Befehl kann auf diese Weise auf alle zugewiesenen Geräte gewechselt werden.

Umlenkung, wofür?

Sinn und Zweck einer Umlenkung kann man am besten mit einigen Beispielen verdeutlichen. Besitzer mit nur einem Diskettenlaufwerk wissen, wovon ich spreche. Wird eine fremde Diskette eingelegt, und man möchte sich das Verzeichnis ausgeben lassen (mit Hilfe des DIR-Befehls), wird immer die Boot-Diskette verlangt, was meistens eine Workbench-Diskette ist. Dies liegt daran, das nach dem Verzeichnis C: gesucht wird, und weil das System weiß, das logische Gerät C: ist auf der Boot-Diskette, wird diese verlangt. Ein umständliches Wechseln der Disketten ist die Folge, und meistens wird zu allem Übel nicht das gewünschte Verzeichnis ausgegeben. Abhilfe schafft zwar die Eingabe

DIR ?

aber es ist nicht der Weisheit letzter Schluß. Der ASSIGN-Befehl kann dienlich sein. Doch bevor dieser in Aktion treten kann, muß das Verzeichnis, das neu zugewiesen werden soll, vorhanden sein. In meinen Beispiel möchte ich das C-Verzeichnis auf die RAM-Diskette umlenken. Dazu muß dieses vorhanden sein, wie wir wissen. Dazu sind folgende Schritte (Befehlsfolgen) vonnöten:

```
MAKEDIR RAM:C
COPY C:#? to RAM:C
ASSIGN C: RAM:C
```

Der erste Befehl erzeugt ein C-Verzeichnis. Der nächste kopiert alle Programme (Befehle), die sich im C-Ordner befinden in das neu erzeugte. Der ASSIGN-Befehl lenkt den Zugriffspfad auf den C-Ordner der RAM-Diskette. Zukünftig wird immer auf die RAM-Diskette zugegriffen, wenn ein Befehl geladen wird. Der Vorteil liegt klar auf der Hand: Die Workbench-Diskette wird bei einem Befehlszugriff nicht mehr benötigt. Ein weiterer Pluspunkt besteht darin, daß die RAM-Diskette natürlich erheblich schneller die Befehle lädt. Ein Nachteil besteht allerdings auch: Die RAM-Diskette belegt natürlich wertvollen Speicherplatz. Besonders bei Rechnern mit nur 512 Kbyte sollten nicht alle Befehle kopiert werden, sondern nur die gebräuchlichsten. Wenn viel freier Arbeitsspeicher zur Verfügung steht, können natürlich auch alle anderen logischen Verzeichnisse auf die RAM-Diskette umgelenkt werden. Hierbei gehen Sie genauso vor, wie zuvor beschrieben.
Wenden wir uns anderen Zuweisungen zu. Zuweisungen von Disketten auf

denen sich Programme befinden, müssen des öfteren vorgenommen werden. Wenn Sie im Besitz einer Festplatte sind, haben Sie vielleicht schon Erfahrung sammeln können. Wollen Sie z.B. ein Textverarbeitungsprogramm, ein Animationsprogramm oder nur ein komplexes Spiel auf die Festplatte kopieren, kommen Sie meistens um die Verwendung des ASSIGN-Befehls nicht herum. Üblicherweise will der Anwender Ordnung auf seiner Festplatte haben und legt zu diesem Zweck für jedes Programm einen eigenen Ordner an. Greifen wir wieder ein Beispiel auf: Das Textverarbeitungsprogramm WordPerfect soll auf Festplatte installiert werden (meistens besitzen größere Programmpakete bereits ein Programm, das die Installation auf Festplatte vornimmt). Die Textverarbeitung soll auf die Festplatte, die mit dem logischen Namen DH0: angesprochen werden kann, kopiert werden. Erzeugen wir zunächst den Ordner, den wir WordPerfect nennen.

```
MAKEDIR DH0:WordPerfect
```

übernimmt das für uns. Der Einfachheit halber kopieren wir alle WordPerfect-Dateien, die sich auf den mitgelieferten Disketten befinden, in das erzeugte Festplattenverzeichnis und löschen die unötigen Dateien anschließend wieder.

```
COPY DF0:#? to DH0:WordPerfect all
```

kopiert zunächst alle Dateien auf die Festplatte. Starten wir aber die Textverarbeitung von der Festplatte, meldet sich das Betriebssystem, daß man gefälligst die original WordPerfect-Diskette in ein Disketten-Laufwerk legen soll und später auch noch die Druckertreiberdiskette. Um diesen lästigen Umstand zu umgehen, muß der ASSIGN-Befehl herangezogen werden.
Die beiden Befehlsfolgen übernehmen die Zuweisung:

```
ASSIGN WordPerfect:DH0:WordPerfect
ASSIGN Printer:DH0:WordPerfect
```

Durch diese beiden Befehlsfolgen wird dem Betriebssystem mitgeteilt, daß der neue Zugriffspfad nicht mehr das Diskettenlaufwerk ist, sondern das neue Verzeichnis auf der Festplatte. Bevor Sie die beiden Zuweisungen machen, müssen Sie unbedingt die Orginaldisketten aus den Laufwerken entfernen, ansonsten wird die Fehlermeldung ausgegeben: "*Can't cancel WordPerfect*". Die beiden Befehlszeilen können Sie ohne weiteres in die Startup-Sequence einfügen, somit ist die Zuweisung immer gegenwärtig und muß nicht erst vorgenommen werden.

Beachten Sie unbedingt, daß der ASSIGN-Befehl keine Zuweisung machen kann, in der ein physikalisches Gerät umgeleitet werden soll. Es sei denn, das Gerät ist nicht angeschlossen oder vorhanden. Beispiel:

```
ASSIGN DF1: DH0:Aztec
```

Existiert das externe Laufwerk, wird unweigerlich eine Fehlermeldung ausgegeben. Sie müssen statt der Laufwerksbezeichnung den Diskettennamen verwenden.

```
ASSIGN Aztec: DH0:Aztec
```

In der neuen Zuweisung hingegen kann oder muß die Gerätebezeichnung verwandt werden.

Ein weiterer großen Nutzen des ASSIGN-Befehls besteht darin, daß ein beliebiges Verzeichnis zu einem logischen Gerät umgewandelt werden kann. Besonders bei Verzeichnissen, auf die der Anwender oft zugreift, die sich aber tief im Verzeichnisbaum befinden, bietet sich eine solche Zuweisung geradezu an.

Beispiel:

```
ASSIGN Q: DH0:Aztec Programme Quelle
```

Die Befehlsfolge weist dem Verzeichnis *'Quelle'*, das sich auf der Festplatte im Ordner *'Programme'* befindet, wobei *'Programme'* im Verzeichnis *'Aztec'* zu finden ist, als logischen Gerätenamen *'Q:'* zu. Der Vorteil liegt klar auf der Hand. Wenn Sie auf den Ordner *'Quelle'* zugreifen wollen, brauchen Sie nicht mehr den kompletten Pfad einzugeben, sondern nur den zugewiesenen Gerätenamen *'Q:'*.

```
CD Q:
```

anstatt

```
CD DH0:Aztec/Programme/Quelle
```

Natürlich können Sie den zweiten Ausdruck auch noch verwenden.
Sie sehen, der ASSIGN-Befehl bietet eine ganze Reihe von Möglichkeiten. Ich hoffe, ich habe mit diesem Artikel ein wenig Licht in den CLI-Dschungel bringen können.

ENDE

```
DF0:
2> assign
Volumes:
Harddisk 1 [Mounted]
TURBOprint [Mounted]
Harddisk 0 [Mounted]

Directories:
print        Harddisk 1:WP
wp           Harddisk 1:WP
T            Harddisk 0:T
S            Harddisk 0:S
L            Harddisk 0:L
C            Harddisk 0:c
FONTS        Harddisk 0:fonts
DEVS         Harddisk 0:devs
LIBS         Harddisk 0:libs
SYS          Harddisk 0:

Devices:
NEWCON DH1 DF2 DF1 DF0
DH0 PRT PAR SER RAW
CON RAM
2>
```

Wird der ASSIGN-Befehl ohne eine Paramaterübergabe aufgerufen, werden alle momentanen Zuweisungen des Systems ausgegeben.

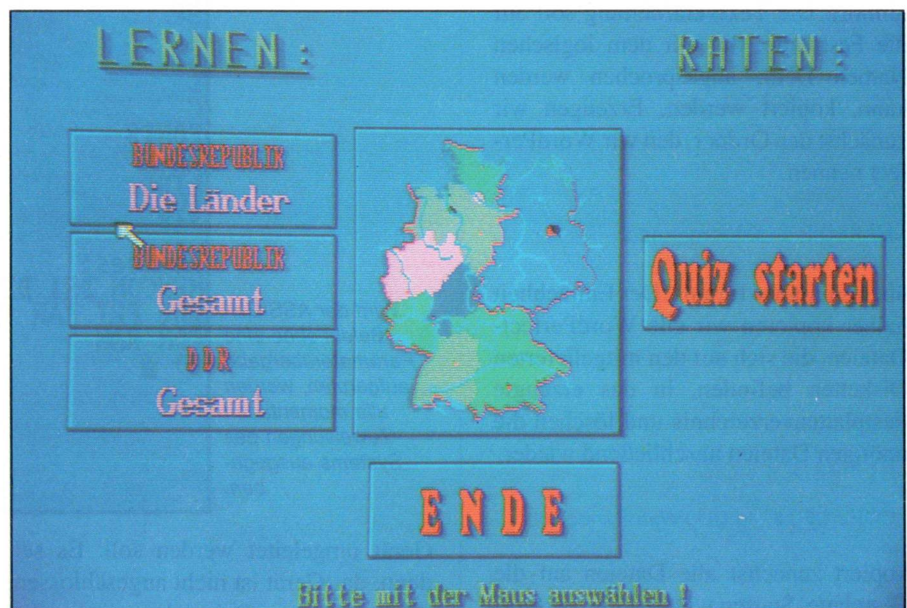
VON ANDREAS KRÄMER

ERDKUNDE

Lernen mit dem Computer

Lernsoftware ist für den AMIGA z.Z. noch rar. Mittlerweile sind aber die ersten Programme dieses Genres erhältlich. Das Programm ERDKUNDE beinhaltet Fragen und Antworten über Städte, Länder, Flüsse und Gebirge.

Das Programm Erdkunde-Deutschland präsentiert sich dem Käufer in einer extravaganten grafischen Verpackung. Inliegend findet sich eine Bootdiskette, eine Ladeanleitung und eine kurze Beschreibung des Programms bzw. der Programmbedienung. Das Programm ist in vier Menüpunkte unterteilt: Länder der Bundesrepublik, gesamte Bundesrepublik, gesamte DDR und Quiz. Durch Anwählen eines der ersten drei Punkte bekommt der Lernwillige Informationen über das betreffende Thema ausgegeben. Der Bildschirm unterteilt sich hierbei in einen Text- und in einem Grafikbereich. Im Grafikbereich wird das betreffende Objekt durch einen optischen Effekt gekennzeichnet. Die Informationen sind zunächst globaler Natur und werden im Laufe spezifischer. Die Bedienung des Lernkurses geschieht ausschließlich mit der Maus. Eingaben über die Tastatur müssen nicht vorgenommen werden.



Das Quiz

Im Quiz kann das Erlernte auf spielerische Art und Weise überprüft werden. Dazu stellt der Computer Fragen, die zuvor im Kurs beantwortet wurden. Vier Antworten stehen zur Auswahl, ein Punktekonto wird ebenfalls geführt. Ist man der Abfrage überdrüssig, kann das Quiz abgebrochen werden. Eine Bewertung wird ausgegeben, die zwischen 'absolut spitze' und 'sehr schwach' gegliedert ist. Das Quiz hat einen Umfang von 250 Fragen, die durch einen bestimmten Zufallsalgorithmus ausgewählt werden.

Fazit

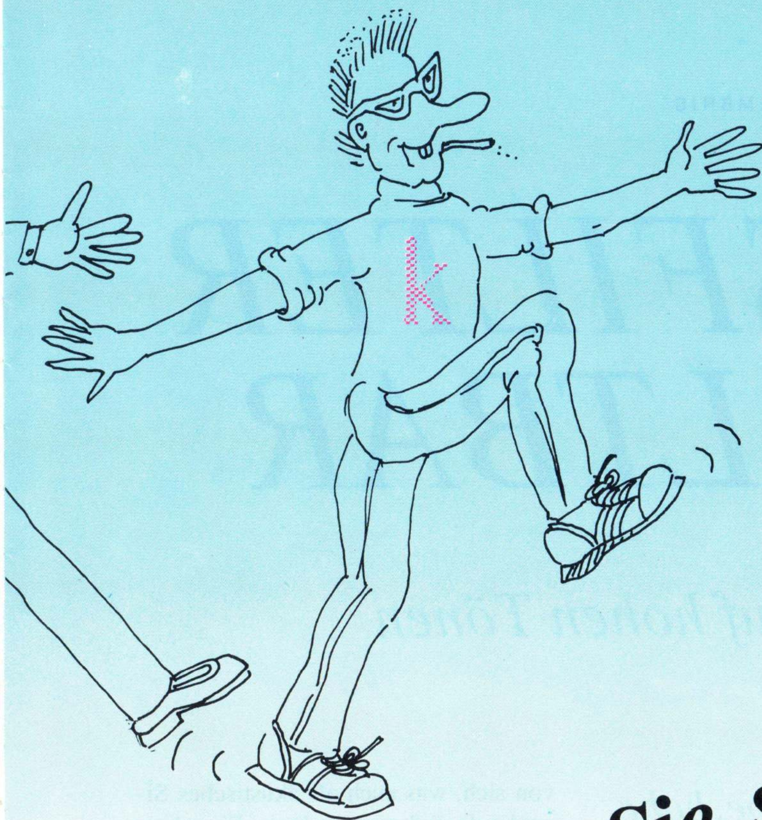
Erdkunde-Deutschland ist eines der wenigen Programme, die den Bereich Lernsoftware aufgreifen, der z.Z. auf dem AMIGA noch sehr vernachlässigt wird. Die Aufmachung des Programms besticht durch viele grafische Effekte, die dem Grafik-Computer AMIGA gerecht werden. Auch mit der Tonausgabe wurde nicht gegeizt, ruhige Musik ertönt beispielweise aus den Lautsprechern während der Lernwillige den Kurs abarbeitet. Didaktisch kann Erdkunde-Deutschland nicht ganz überzeugen, zu sehr erinnert die Aufmachung an ein Buch, für das Medium Computer gibt es hingegen bessere und effizientere Lernmethoden.

Für 49.90 DM ist das Produkt aber geeignet, um einmal den Joystick aus der Hand zu legen. Das Allgemeinwissen bezüglich der Bundesrepublik und der DDR kann enorm aufgebessert werden. Auch AMIGA-Anwender, die glauben, Ihr Erdkundewissen sei ausreichend, werden noch allerlei Wissenswertes erfahren. Neben Erdkunde werden auch eine Menge geschichtliche Hintergründe an den Anwender weitergegeben.

Anbieter: CDC
Louisenstr. 115
6380 Bad Homburg
Tel. 06172-24748

Preis: 49.90 DM

ENDE



AMIGA

K. Schneider / O. Steinmeier

GRUNDLEHRGANG

Der richtige Einstieg

Heim Verlag

Holen Sie sich auch den neuen AMIGA-GRUNDLEHRGANG

DM 59,-

Buch und Diskette

unverbindlich empfohlener Verkaufspreis

WICHTIGE MERKMALE:

★ Das Buch für den richtigen Einstieg mit dem Commodore AMIGA ★ Auf über 400 Seiten werden dem Leser leicht verständlich die Grundlagen der Computertechnik und der Umgang mit Hardware erklärt ★ Ein ausführlicher Hauptteil ist dem Einsatz der grafischen Benutzeroberfläche des Betriebssystems gewidmet. Hier erläutert das Buch Fenster, Pulldown-Menüs und die vielen anderen Teile der ***Workbench*** ★ Wer die Maus nicht mag, der kann aus dem Kapitel über den ***Command Line Interpreter (CLI)*** entnehmen, wie man den AMIGA auch ohne Maus einsetzen kann ★ Ein weiterer Bereich des Buches ist die Einführung in die Programmiersprache ***BASIC.*** Eine umfangreiche Befehlsübersicht sowie einige interessante Programme dienen der Erlernung und dem guten Training von BASIC ★ Anhänge wie z. B. ein ***Index*** und eine ***Sachworterklärung*** bieten das schnelle Nachschlagen und Auffinden wichtiger Punkte ★ ***Mit dem Buch erhalten Sie eine Programmdiskette mit allen abgedruckten Listings.*** Damit können die Beispielprogramme ohne die Mühe und Arbeit des Eintippens auf dem Computer nachvollzogen werden.

AUS DEM INHALT:

1. Die Hardware des AMIGA
★ die versch. AMIGA-Modelle ★ die Diskettenstation ★ Anschluß eines Druckers ★ Monitore am AMIGA ★ Erweiterung des AMIGA-Systems ★ Einstieg in die MS-DOS Welt mit dem AMIGA ★ Die „Innereien" des AMIGA (RAM, ROM u. Prozessoren)

2. Das Betriebssystem des AMIGA
★ Betriebssysteme und ihre Bedeutung ★ Die Benutzeroberfläche des AMIGA ★ Steuerung der Workbench ★ Arbeiten mit Maus, Fenstern und Pull-down-Menüs ★ Verwendung von Disketten, Dateien, Directory ★ Die Programme der Workbench Diskette im Einzelnen ★ Der CLI und seine Bedienung ★ Kopieren, Löschen und Batch-Bearbeitung im CLI

3. Programmieren in Amiga-Basic
★ Die Bedienung des Basic-Interpreters ★ Variable in Basic ★ Schleifenstrukturen ★ Die IF-Abfrage ★ Proceduren zur Programmstrukturierung ★ Graphik-Programmierung in AMIGA-BASIC ★ Dateiverwaltung ★ ausführliche Befehlsübersicht mit detaillierten Erklärungen

4. Zum Training
★ Programm-Diskette mit allen abgedruckten Listings ★ Sachworterklärung (Fachwörter-Lexikon) ★ Ausführlicher Index (Stichwortverzeichnis mit entspr. Verweisen)

BESTELL-COUPON an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Ich bestelle ____________________
zzgl. DM 5,– Versandkosten (unabhängig von der bestellten Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ____________________

Straße, Hausnummer ____________________

PLZ, Ort ____________________

Benutzen Sie auch die in KICKSTART vorhandene Bestellkarte

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

VON DANIEL GEMBRIS

TIEFPASSFILTER ABSCHALTBAR

Der AMIGA pfeift auf hohen Tönen

Der Tiefpaßfilter im AMIGA hat die Aufgabe, hohe Töne (ab 4kHz), häufig Störfrequenzen, abzuschwächen bzw. abzuschneiden (ab 7kHz). Doch leider ist dies dem Klang nicht immer dienlich. Unsere kleine Schaltung schafft Abhilfe.

Eine Nebenwirkung des Tiefpaßfilters ist die Eigenschaft, Rechtecksignale zu deformieren. Dazu muß man wissen, daß Rechtecksignale aus sehr vielen Einzelfrequenzen zusammengesetzt sind. Werden die hochfrequenten Obertöne durch den Filter abgeschnitten oder begrenzt, bleibt im Extremfall nur ein Sinussignal übrig. Beim A500/A2000B läßt sich der Tiefpaßfilter ja glücklicherweise über die Power-LED ausschalten. Die hier vorgestellte Schaltung macht dies auch beim A1000/A2000A möglich.

Die Theorie

Abb.1 zeigt den Schaltplan der Schaltung. Um sie verstehen zu können, muß man zunächst wissen, daß PA1 der CIA-A die Power-LED indirekt ansteuert. Zwischen PA1 und der LED liegt nämlich noch eine Schaltung, die den Pegel invertiert und dafür sorgt, daß die LED nicht aus-, sondern nur dunkel geschaltet werden kann.

Wenn die Power-LED hell leuchtet, liegt PA1 auf low. Wenn die Power-LED schwach leuchtet, ist PA1 logisch high. Dadurch schaltet der Transistor T1, dessen Basis an PA1 über den 470 Ohm-Widerstand angeschlossen ist, die Collector-Emitter-Strecke durch. Nun fließt Strom (Spannung 12V) über den 100 Ohm Widerstand und die Relaisspule nach Masse. Die Diode dient dazu, Störspannungen, die sehr hoch sein können, abzubauen. Die Relaisspule schließt die 2 Schalter, die Pin 8 und Pin 5 sowie Pin 14 und Pin 3 des Verstärker-ICs über jeweils einen 22 kOhm-Widerstand kurzschließen. Die beiden Tiefpaßfilter (einer für den linken, der andere für den rechten Tonkanal) sind somit überbrückt. Das wird rückgängig gemacht, sobald die Power-LED wieder hell leuchtet.

Die Praxis

Wenn Sie ein anderes Relais (Ein Relais gibt beim Umschalten noch einen Laut von sich, was auch als akustisches Signal aufgefaßt werden kann. Eine digitale Lösung wäre auch möglich gewesen, würde aber aus dem angegebenen Grund nicht gewählt) als in der Liste aufgeführt verwenden möchten, sollte die Schaltung auf einer Lochrasterplatine aufgebaut werden. Sonst auf einer geätzten Platine. Die Reihenfolge, in der die Bauelemente gelötet werden, spielt keine Rolle.

Nun muß der Computer geöffnet werden. Beachten Sie bitte, daß dadurch eine eventuell noch bestehende Garantie erlischt. Das 5-adrige Kabel muß am Verstärker-IC angelötet werden. Man sollte nur kurz am IC löten. Es ist nämlich nicht gesockelt und ließe sich somit bei einem Defekt nur schwer auswechseln. Die Lage des Verstärker-ICs können Sie schnell mit Hilfe von Abb.2 ermitteln. Die Kabelenden werden an Pin 8, Pin 5, Pin 14, Pin 4 und Pin 3 angelötet. Jetzt muß noch die CIA-A, deren Lage Sie ebenfalls mit Abb.2 ermitteln, vorsichtig mit einem Schraubenzieher aus dem Sockel gehebelt, der Sockel, der mit der Schaltung über das 2-adrige Kabel verbunden ist, eingesetzt und in diesen Sockel die CIA eingesteckt werden. Es ist sinnvoll, der Schaltung ein Gehäuse zu spendieren,um Kurzschlüsse zu vermeiden. Das Gehäuse sollte mindestens 2 cm hoch, 6 cm lang und 4 cm breit sein.

Bevor Sie den Rechner wieder schließen, sollten Sie überprüfen, ob die Ver-

bindung der Schaltung mit dem Verstärker-IC und der CIA in Ordnung ist. Sie können dazu auch den Schaltplan (Abb. 1) zu Rate ziehen. Nun kann der AMIGA eingeschaltet werden. Falls sich nur ein grüner Bildschirm zeigt, haben Sie entweder vergessen, die CIA wieder im Rechner unterzubringen oder sie steckt verkehrtherum im Sockel, oder ein Beinchen steckt nicht in der Fassung. Ob ein Kurzschluß vorliegt, kann dadurch ermittelt werden, daß der Widerstand zwischen +5 V und Masse und +12 V und Masse gemessen wird. Wenn der Widerstand annähernd 0 Ohm beträgt (genau 0 Ohm kann er nicht betragen, da die Meßkabel auch einen Widerstand haben), handelt es sich einwandfrei um einen Kurzschluß. Wir gehen davon aus, daß es Ihnen nach einer gewissen Zeit gelingen wird, nach dem Einschalten des Rechners die Hand auf dem Bildschirm erscheinen zu lassen. Um zu testen, ob die Schaltung läuft, starten Sie den BASIC-Interpreter, geben das kleine BASIC-Programm ein und starten es. Der zweite Ton müßte höher sein als der erste und der vierte höher als der dritte. Wenn nicht, muß die Platine der Schaltung auf Leiterbahnunterbrechungen hin untersucht und überprüft werden, ob alle Bauelemente richtig in der Platine stecken, und, ob es keine kalten Lötstellen gibt. Falls der Test zur vollsten Zufriedenheit ausfallen sollte, können Sie sich an weiteres Ausprobieren der neuen Hardware machen. Es gibt etliche neue Anwendungen, die durch diese Schaltung möglich werden:

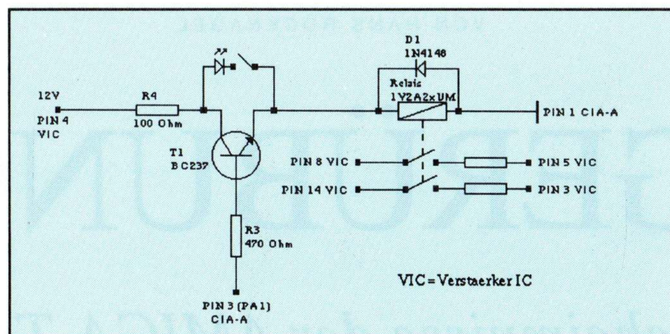


Abb. 1: Der Schaltplan der kleinen Schaltung

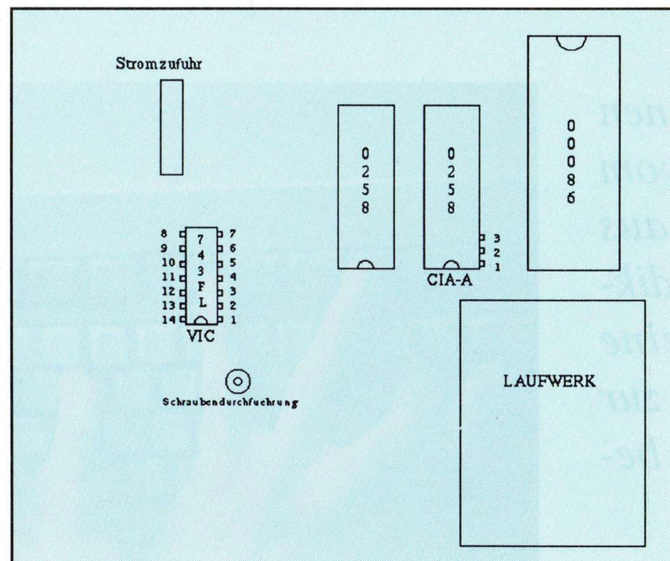


Abb. 2: Die betreffenden ICs müßten mit Hilfe der Skizze im AMIGA 1000 leicht zu finden sein.

- High Frequency Sampling
- Erzeugung von Ultraschall (vielleicht zum Verjagen von Mücken?)
- Erzeugung von HF bis 1 MHz!! (das geht aber nur mit einem Trick)

.....

Wenn Sie z.B. folgendes Assemblerprogramm vor dem Starten eines Spiels, das eine tolle Titelmusik hat, aufrufen, wird diese oft gleich noch besser klingen:

```
MOVE.B  #2,$BFE001
RTS
```

Mit dem kleinen Programm wird der Filter ausgeschaltet, das zweite schaltet den Filter wieder an.

```
MOVE.B #0,$BFE001
RTS
```

Wenn Sie keinen Assembler besitzen, können Sie die beiden BASIC-Hexdump-Programme eingeben. Sie erzeugen die ausführbaren Programme.

Eine weitere Möglichkeit besteht darin, den Tiefpaßfilter direkt ein- bzw. auszuschalten. Dazu müssen Sie lediglich die Emitter-Kollektor-Strecke des Transistors überbrücken. Über einen Schalter können Sie die Verbindung herstellen oder unterbrechen.

ENDE

Bestückungsplan:

1 Diode 1N4148
1 Transistor BC237 oder ähnliche
1 Widerstand 100 Ohm
1 Widerstand 470 Ohm
2 Widerstände 22 KOhm
1 Relais 6V, 2A, 2xUM
1 Fassung 40-polig
50 cm 5-adriges Flachbandkabel
50 cm 2-adriges Kabel

Zum direkten Schalten des Filters:

1 Schalter 1x ein/aus
1 Leuchtdiode

```
WAVE 0,SIN
WAVE 1,SIN
POKE 12574721&,0
SOUND 15000,50
POKE 12574721&,2
SOUND 15000,50
POKE 12574721&,0
SOUND 15000,50,,1
FOR t=1 TO 1000:NEXT t
POKE 12574721&,2
SOUND 15000,50,,1
```

Das kleine Programm testet die Funktion des Filters.

```
' filter_aus Lader

Filename$= "filter_aus"
OPEN "O",1,Filename$
FOR I=0 TO 47
  READ Wert
  PRINT #1, CHR$(Wert);
NEXT I
CLOSE(1)

DATA 0,0,3,243,0,0,0,0,0,0
DATA 0,1,0,0,0,0,0,0,0,0
DATA 0,0,0,3,0,0,3,233,0,0
DATA 0,3,19,252,0,2,0,191,224,1
DATA 78,117,0,0,0,0,3,242
```

BASIC-Hexdump zum Ausschalten des Filters.

```
' filter_an Lader

Filename$= "filter_an"
OPEN "O",1,Filename$
FOR I=0 to 47
  READ Wert
  PRINT #1, CHR$(Wert);
NEXT I
CLOSE(1)

DATA 0,0,3,243,0,0,0,0,0,0
DATA 0,1,0,0,0,0,0,0,0,0
DATA 0,0,0,3,0,0,3,233,0,0
DATA 0,3,19,252,0,0,0,191,224,1
DATA 78,117,0,0,0,0,3,242
```

BASIC-Hexdump zum Einschalten des Filters.

VON HANS RÖCKNAGEL

FINGERÜBUNGEN

Die Geheimnisse der AMIGA-Tastatur

Leider muß ich Ihnen diese Zeilen vom Krankenbett aus schreiben resp. diktieren, da meine Arme und Hände zur Zeit noch nicht benutzbar sind...



Wie alle Amiga-Besitzer fahndete auch ich nach den geheimen Nachrichten, die sich im Amiga-Betriebsystem verstecken. Eine der lächerlichsten Übungen - Sie kennen sie - ist das gleichzeitige Drücken der Amiga- ,der Shift- und einer der Funktionstasten, die dann mehr oder weniger interessante Nachrichten der Betriebssystementwickler auf den Bildschirm bringen. Ich sagte mir, daß diese genialen Leute bestimmt informative, freche, ja vielleicht gar erotische Meldungen implementiert haben! Ich machte mich also in schlaflosen Nächten auf die Suche nach den wirklichen Supermeldungen - und hier sind sie, die genauen Anweisungen, um eine Top-Meldung auf den Schirm zu kriegen ...!

Schritt 1:
Rufen Sie Preferences auf. Klicken Sie sich durch das Menü, bis die Threshold-Schiene erscheint. Klikken Sie nacheinander, und zwar in einem Abstand von genau 2 Minuten und 16 Sekunden, folgende Werte an:

3, 6, 2, 9, 12, 4, 13, 9, 5, 9, 9, 9, 9, 15, 2, 7, 6, 8, 9, 9, 9, 9, 14, 5, 3, 8, 1, 1, 1, 1, 1, 1.

Gehen Sie mit dreimaligem Cancel auf die Workbench und rufen Sie SAY auf.

Schreiben Sie:

"Jay, what a famous computer! He speaks!"

Der Rechner gibt darauf eine akustische Meldung aus.

Schritt 2:
Booten Sie den Amiga und halten dabei folgende Tasten gedrückt:

- **Die geshiftete Tab-Taste**
- **Amiga-Taste links**
- **F10**
- **die PrtSc-Taste auf dem Zehnerblock**
- **die Backspace- und die Ä-Taste**

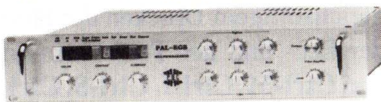
(es empfiehlt sich eventuell die Einladung einer verständnisvollen Freundin ...!).

Falls der Amiga tatsächlich gebootet hat, ist nun folgendermaßen vorzugehen: Sie entfernen siebenmal hintereinander die Diskette aus dem Laufwerk df0:. Wenn Sie sie zum achten Mal hineinlegen, müssen Sie gleichzeitig ESC, Enter am Zehnerblock, F6 sowie die Space-



Peter Biet Computerdesign
Georg-Fischer-Str. 5, 6415 Petersberg 2
Telefon 06 61 / 60 12 63





PAL-RGB-MULTIPROZESSOR 2000 (Version I, II und III)
Die neue Gerätegeneration, mit noch mehr Einsatzmöglichkeiten. Durch die enorme Vielseitigkeit verknüpft mit guten Leistungsmerkmalen und einem bisher unschlagbaren Preis-Leistungs-Verhältnis stellt dieses neukonzipierte Gerät einen Meilenstein im Bereich der Desktop-Videobearbeitung mit dem Amiga dar.

Einsatzbereiche:

1.) Elektronischer **RGB-Farbsplitter** (automatisch und manuell!!). Separate Regelmöglichkeiten in den Bereichen **Helligkeit, Kontrast, Farbsättigung** und die einzelnen **RGB-Intensitätswerte**.
Damit ist es auch mit **DIGI-VIEW** und einer Farbcamera oder einem Videorecorder möglich, absolut professionelle Ergebnisse zu erzielen!
2.) **Video-Colorprozessor:** dient zur Veränderung bzw. Verfremdung des anliegenden Farbvideosignals in o. g. Bereichen zwecks Überwachung, Kontrolle und Überspielung
3.) **Videoüberspielverstärker:** mit Level und Konturregelung.
4.) **RGB-PALUMWANDLER:** erzeugt aus dem Computer-RGB-Signal ein PAL-Videosignal. Ebenfalls in o. g. Bereichen regelbar! (Ersetzt damit alle herkömmlichen PAL-Videokarten.)

neu 19 Zollgehäuse mit neuem Design, Digitastern, Anzeigendisplay und Chinchbuchsen.
neu Mehrfachüberspielmöglichkeit auf bis zu drei Videorecorder gleichzeitig.
neu **Anschlußmöglichkeiten eines Multisynchronmonitors als Videomonitor über RGB-Ausgang.**
neu Einspeisung des neuen **Super VHS** Signals und Umwandlung in ein normales FBAS-Signal. (Vollständig regelbar ohne Qualitätsverlust!)
neu DIGI-VIEW-Digitizer kann direkt intern eingesteckt werden. (Stromversorgung durch eigenes Netzteil, keine externe Kabelverbindung mehr nötig und Druckerumschaltmöglichkeiten!) **Preis auf Anfrage**

DVS-2000 System
Das bewährte Realtimefixing Digitizersystem in voller PAL- und Overscanauflösung.
Lieferumfang:
- **Digitizer** (alle Auflösungen incl. HAM 2-4096 Farben in PAL!)
- **Digitaler VHS-Videorecorder** (eigenständige Nutzung möglich!) (kann auch durch einen Super-VHS-Recorder ersetzt werden!)
- **PAL-RGB-MULTIPROZESSOR 2000** Version III

zusammen nur 2598,– DM (auch einzeln)

Demo-Dia-Show (DVS-2000 und PAL-RGB-MULTIPROZESSOR 2000) **10,– DM**

PAL-RGB-GENLOCK
Das neue Multifunktionstalent mit eingebautem Audiomischer und **Fernbedienungseinheit.** Separate RGB- und FBAS-Ausgänge. Stufenloses Ein- bzw. Ausblenden von Computer- und Videosignalen über separate Regler möglich. **nur 998,– DM**

NEU: Panasonic Super-VHS-Videorecorder (Hifi + Nachvertonung!) **2998,– DM**
NEU: Panasonic Camcorder MC-10 hohe Bildauflösung! **3198,– DM**

Norbert Dornhöfer & Michael Böttcher GbR

DFÜ-SHOP

Discovery 1200 C+
Hayes-Kompatibles 1200 Baud-Modem, 300, 1200 Baud Vollduplex, CCITT (V21, V22), Bell (103, 212A), eingebauter Lautsprecher, Auto-Dial, -Answer, -Redial u.v.m.
Preis inkl. Steckernetzteil nur **DM 279.–**
Discovery 1200H (PC/XT-Karte) **DM 198.–**

Discovery 2400 C
2400 Baud-Modem, sonst wie 1200C+ **DM 449.–**

Ext. ATARI-Drive (NEC 1037A, abschaltbar)	DM 279.-
Ext. AMIGA-Drive (NEC 1037A, abschaltbar, Bus)	DM 259.-
AMIGA-Drive (NEC 1037A, f. A2000, intern)	DM 229.-
VIRUS FORSCHER SET	DM 39.-

PUBLIC-DOMAIN-SERVICE
BERLIN
Trackanzeige DM 79.–

DFÜ-SHOP Norbert Dornhöfer & Michael Böttcher GbR Kolonnenstraße 33
1000 Berlin 62 • Tel.: 030 782 71 18
(★) Anschluß ans öffentl. Telefonnetz ist strafbar!



Public-Domain

SUPERPREISE!!

Über 2000 Disks im Archiv!

Jede 3,5" PD-Disk. . . . . 2,70 DM
ab 50 Stück nur **2,60 DM**

Wir verwenden nur errorfreie Qualitätsdisketten!

Wir liefern: Fish, Chiron, RPD, Poseidon, Kickstart, Auge, Tornado, Panorama, Bordello, Amicus, Faug, Ruhr, Cactus, ACS, Taifun, Franz, RHS, PornoShow, TBAG, SACC und ca. 25 andere Serien !

Leerdisketten 3,5" 2DD 135 TPI 10 Stück DM 21.-

Spielepaket 10 Disketten=ca.40 PD-Spiele DM 40.-
Einsteigerpaket für Amiga-Anfänger (Utilities, CLI-Hilfen, Infos usw.) 10 Disk. DM 40.-
Das Superpaket bestehend aus Textverarbeitung, CAD, Haushaltsprogramm, Anti-Virus-Disk usw. alles mit deutscher Anleitung! 15 Disk. DM 55.-
Das Soundpaket 10 Disketten mit tollen Sonix-Super-Sounds, inkl. Sonix-Player-Disk DM 40.-
Das Super-Mix (je 5 Disketten aus vorgenannten Paketen) 20 Disk. zum SONDERPREIS von nur DM 69.-

Alle Preise zuzüglich Versandspesen.

Wir suchen laufend Programme aller Art ! Rufen Sie an !

Tel. 05202/71099

ABC-SOFT
I. Güldenpfennig, Postfach 1124, 4811 Oerlinghausen

Transfile Amiga, Amiga 1600, Amiga 850

„Die perfekte Kopplung."

NEU! nun auch für Casio FX 850

TRANSFILE Amiga koppelt Ihren Sharp-/Casio-Pocketcomputer mit Ihrem Amiga. Dies ermöglicht Ihnen nicht nur das sichere Abspeichern von Daten und Programmen auf Diskette, sondern auch die sichere Übertragung in beide Richtungen sowie das Editieren und Drucken der Daten.
Für weitere Produktinfos rufen Sie uns einfach an.

Komplett mit Interface, Diskette und Anleitung

DM 129,–

Händleranfragen erwünscht.

yellow C·O·M·P·U·T·I·N·G
Postfach 1136/3
D-7107 Bad Friedrichshall
Telefon 0 71 36 / 2 00 16

Speicherprobleme mit dem Amiga 2000?
Wir bieten die Lösung!

- Eine Grundplatine, wahlweise mit 2, 4, 6 oder 8 Megabyte bestückt.
- Jederzeit problemlos auf 8 Megabyte nachrüstbar
- Abschaltbar Autokonfigurierend
- O-Wait-State

2 Megabyte:	1498,- DM
4 Megabyte:	2998,- DM
6 Megabyte:	3998,- DM
8 Megabyte:	4998,- DM

AB LAGER!!!

Händleranfragen erwünscht!

Ralf Jochheim Computer Tuning
Binsengrund 22, 2000 Hamburg 70
Telefon 040/6956718

Taste betätigen. Drücken Sie nun - sehr wichtig! - sofort beim Erscheinen des Disketten-Icons dreimal abwechselnd die linke und die rechte Maustaste!
Spätestens jetzt wird Ihre Freundin eine sehr interessante Meldung ausgeben ...!

Schritt 3:
Die letzte Unternehmung sollte mich kurzfristig ins Krankenhaus bringen...
Die Programmierung des Amiga-Betriebssytems muß schon eine harte Sache gewesen sein. Als Basic-Programmierer kann man das schon nachfühlen...
Ich dachte mir deshalb, daß es vielleicht eine Systemmeldung geben müßte, die den Verzweiflungsschrei eines der Entwickler in etwa wiedergibt! Vielleicht in Form eines kleinen Verses! ("I'm so young and beautiful but Irwin don't let me free!")

Ich machte mich nun intensiv an die Arbeit und dokumentiere sie hier im folgenden:

- **Booten des Amiga von df0:**
- **Öffnen des WB-Fensters mit normalem Doppelklick unter gleichzeitigem Drücken der F1-, der F6-, und der 3er-Taste auf dem Zehnerblock.**
- **Unter Beibehaltung dieser Tastendruckkonfiguration die WB-Diskette aus dem Laufwerk nehmen und die ExtrasD-Diskette einlegen. Deren Fenster normal öffnen und alle Icons mit Shift-Mausklick aktivieren. Anschließend die ExtrasD-Disk aus dem Laufwerk entfernen und eine unformatierte Diskette der Firma db einlegen.**
- **Das Edit-Menü aufrufen und unter Festhalten der Del-Taste Snapshot aktivieren.**

Nachdem ich noch versucht hatte, das Basic-Programm "Dots" unter gleichzeitigem Drücken der Help-, der Esc- und der Tastenkombination **Ctrl-M, Ctrl-I, Ctrl-S, Ctrl-T** zu starten, mußte mich mein einzig noch verbleibender Bruder davon abhalten, die sicherlich erfolgreiche Schlußsequenz "Sechsfacher Doppelklick rechts im 0.3 Sekundentakt unter Einbeziehung der Tastendruckfolge **Alt-#, Alt-Alt, Alt-Alt** andersherum, **Alt-Ctrl, Ctrl-Alt** und **Amiga-links/rechts-Alt-ß-Shift-Enter** auszuprobieren.

Mein Bruder beförderte mich ins Krankenhaus, wo eine akute Armverwindung rechts/links, sowie beginnende Krampfzustände beider Hände diagnostiziert wurden.

Die Geheimnisse des Amigas fordern eben ihren Preis!

ENDE

A. Dimou, Hard- u. Software-Versand
Baumstr. 8 · 7536 Ispringen · Tel. (0 72 31) 8 16 93

- Amiga 500 + RGB Monitor + Epson LQ 500 + Maus + 10 PD-Disks DM 2.498,00 frei Haus
- Amiga 2000 + RGB Monitor + Epson LQ 500 + Maus + 10 PD-Disks DM 3.399,00 frei Haus
- Amiga 2000 + RGB Monitor + Epson LQ 500 + XT-Karte + Maus + Software + 10 PD-Disks DM 4.299,00 frei Haus
- Festplatten .... ab 899,00 DM
- Nec P 2200, 24 Nadeln, deutsche Version .... 899,00 DM
- Epson LQ 500, 24 Nadeln, deutsche Version ...... 899,00 DM
- 3 1/2 Zoll Disketten NN 2 DD .... ab 2,59 DM
- 5 1/4 Zoll Laufwerk, abschaltb., Port durchgeschliffen DM 339,00
- 3 1/2 Zoll Laufwerk, abschaltb., Port durchgeschliffen DM 270,00
- PD-Bücher I/II je DM 49,00 PD-Disks zu Bücher je 99,00 DM

Wir führen auch anspruchsvolle Strategie Spiele.

Wir liefern per Nachnahme. Für herstellerbedingte Lieferverzögerungen übernehmen wir keine Haftung. Alle Angebote freibleibend.

Musik- und Grafiksoftware Shop
Das Spezialgeschäft für Midisoftware

Wasserburger Landstr. 244 ★ 8000 München 82 ★ Tel.: 0 89/4 30 62 07

BRANDNEU!!! – 'THE QUEST SEQUENZER'
für AMIGA!!!

das neue AMIGA-Profi-Sequenzerprogramm zum Sensationspreis von DM 298,–

Bei diesem Sequenzerprogramm wurde besonderer Wert auf exaktes Timing gelegt, sodaß nun auch ein professioneller Einsatz z. B. im Studio mit dem AMIGA möglich geworden ist.
"The Quest Sequenzer" ist der Erste einer dreistufigen Sequenzerreihe. Damit haben Sie die Möglichkeit, ein leicht zu bedienendes Basisprogramm zu erwerben, daß Sie dann später je nach Bedarf auf die mittlere oder höchste Stufe ausbauen können. Sie zahlen dafür nur den entsprechenden Differenzbetrag. Durch diesen Service wird der Einstieg für ein perfektes MIDI-Studio geradezu ideal.

MIDI-Editorsoftware AMIGA für:
Yamaha DX/TX Serie | Casio CZ Serie
Roland MT-32, D-10/20/50/110 | Ensoniq ESQ/SQ Serie

Fordern Sie unsere kostenlose AMIGA-MIDI-LISTE an.
Unsere Fachleute beraten Sie gerne in allen Fragen zum Thema "MIDI mit dem Computer"

Rufen Sie uns einfach an oder besuchen Sie uns in unserem Laden!
Mo–Fr 10–18.30 Uhr Sa 9–13.00 Uhr

HuBCom
Hochhuth und Broschard GbR
Computerartikelversand

Jens Hochhuth
Lange Rötterstr. 7
6800 Mannheim
Tel.: 06 21 / 37 76 16
Btx: 06 21 / 37 34 31



Thomas Broschard
Karlsruher Str. 19
6940 Weinheim
Tel.: 0 62 01 / 1 76 08
Btx: 0 62 01 / 18 24 63

HuBCom 31 MB Festplatten für AMIGA 500/1000, kompatibel zu gängigen Speichererweiterungen, amigafarben DM 1079,–
HuBCom 31 MB Festplatten für AMIGA 2000 DM 899,–
Amiga 500 + Monitor 1084 Stereo DM 1499,–
Neu: Btx für Amiga (Hard- und Software) DM 199,–

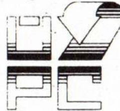
Public Domain Software, z. B. Fish –164, RPD –142, RHS –85, Kickstart –75, Faug –51, Chiron Con. –101, RW –14, Ruhr –15...
Über 1000 PDs auf 3.5" 2S/2D Verbatim Markendisks ab DM 3,20
2 Infodisketten DM 6,– Vorkasse, DM 8,– Nachnahme
Sonderangebot 100 PDs Ihrer Wahl auf 3.5" 2S/2D Verbatim Diskette für 320,–

PROBLEME MIT ENGLISCHER ANLEITUNG?
Übersetzen von englischen Bedienungsanleitungen, Computerhandbüchern und Programmen für AMIGA, Atari, Macintosh, MS-DOS, Unix-Xenix, Periferiegeräte und andere durch *Spezialisten.*

Sonderangebote deutscher Anleitungen für AMIGA

Aegis Sonix	39,–	**DBW Render**	20,–
Flight Simulator II	29,–	**Dr. Ts' KCS**	49,–
Aegis Draw plus	49,–	**Grabbit**	20,–
Aegis Impact	39,–	**CLI-Mate**	20,–
Aegis Audiomaster	39,–	**PrtDrvGen Druckertreibergenerator**	15,–
Sekaassambler	29,–	**Galileo**	29,–

Bei Versand im Inland berechnen wir DM 5,– für Porto Verpackung/NN.
Versand ins Ausland nur mit Vorauskasse + DM 4,– für Porto/Verpackung



0 83 74 - 98 73

T. Sonnenmoser · Hauptstraße 26 · D-8961 Haldenwang

Btx/Vtx-Manager

Btx/Vtx.
Jetzt auf dem Amiga!

Die neue Welt der Telekommunikation läßt sich mit dem *Btx/Vtx Manager* komfortabel nun auch auf dem Amiga handhaben. Informationen über dieses „Fenster" zur qualifizierten, maßgeschneiderten Information senden wir Ihnen auf Anfrage gerne zu.

Drews EDV + Btx GmbH
Bergheimerstraße 134 b
D-6900 Heidelberg
Telefon (0 62 21)
2 99 00 und 2 99 44
Btx-Nummer 0622129900
Btx-Leitseite *2 99 00 #

d
D r e w s

SKy-Ware
Amiga-Zubehör von Spezialisten

Game-Trainer:
Setzt den Prozessor-Takt bis auf Null herab.
Lötfreier Einbau in A500/1000/2000 49,- DM

Multifunktions-Karte für AMIGA-PC/Side-Car:
Accu-Uhr, Ser.-Port, Par.-Port, incl. Software
Speichererweiterung auf 640 KB-RAM 299,- DM

Externes 3,5" Laufwerk
durchgeschleifter Bus, abschaltbar, 40/80 Spuren
Metall-Gehäuse, Bootfähig!! 249,- DM

AMIGOS-Festplatte 20 MB
original Amigos-Festplatte incl. Netzteil, im Stahlgehäuse. Gehäuse paßt unter Monitor.
Anschluß am Expansions-Port (durchgeschliffen).
Incl. Software & WB 1.3. 968,- DM

File-Card für PC-Karte/Side-Car
20 MB: 779,- DM 30 MB: 929,- DM 50 MB: 1398,- DM

SKY-WARE Peter Engels
Postfach 13 31 · 5308 Rheinbach
Tel: 0 22 26 / 57 14 o. 0 22 53 / 25 18

AZTEC-C FÜR AMIGA VERSION 3.6

Wußten Sie, daß eines der verbreitesten und komplexesten Betriebssysteme - UNIX - in C geschrieben ist ?

Wußten Sie, daß auch das Betriebssystem des AMIGA größtenteils in C geschrieben wurde ?

Wußten Sie, daß C eine der wichtigsten und modernsten Programmiersprachen ist ?

Wußten Sie, daß Aztec-C einer der schnellsten und leistungsfähigsten Compiler für den AMIGA ist ?

Wußten Sie, daß jetzt Aztec-C in der Version 3.6 verfügbar ist ?

Möchten Sie mehr darüber wissen? Dann schicken Sie uns einen ausreichend frankierten Rückumschlag und Sie erhalten ausführliche Information.

Up-Date-Service für alle MANX-Kunden auch bei uns. Fragen Sie nach!

HIERMIT BESTELLE ICH:

- ☐ AZTEC-C68K/AM-P PROFESSIONAL SYSTEM FÜR DM 298.-
- ☐ AZTEC-C68K/AM-D DEVELOPER SYSTEM FÜR DM 598.-
- ☐ AZTEC-SDB SOURCE LEVEL DEBUGGER FÜR DM 149.-

Versandkosten: Inland DM 7,50 Ausland DM 10,-
Auslandbestellungen nur gegen Vorauskasse
Nachnahmegebühr DM 3,70

- ☐ Vorauskasse
- ☐ Nachnahme

NAME: ____________

VORNAME: ____________

STRAßE: ____________

ORT: ____________

UNTERSCHRIFT: ____________



MERLIN COMPUTER GMBH
INDUSTRIESTRAßE 26
6236 ESCHBORN
TEL. 06196/481811

VON JÖRG WILHELM / WOLF DIETRICH

DIE NÄCHSTE GENERATION

Motorola 68030 und Inmos T800 - Zwei Prozessoren mit Zukunft



Traurig, aber wahr: von der technischen Seite betrachtet, ist der Hauptprozessor des AMIGA, der Motorola 68000, bereits veraltet (wenngleich er in der täglichen Anwendung doch noch überzeugen kann). Heutzutage, wo 32 Bit Busbreite selbstverständlich sind, ringen verschiedene neuentwikkelte Prozessoren um die Gunst des Anwenders (oder besser, der Computerhersteller). Dabei sind die Chip-Konstrukteure verschiedene Wege gegangen: Während Motorola mit dem 68030 eine klassische, zum 68000 und 68020 weitgehend kompatible, aber dennoch stark optimierte CPU präsentiert, entwickelte INMOS mit dem T800 einen Prozessor, der auf einem anderen Prinzip, dem inzwischen wohl allgemein bekannten Transputerkonzept, basiert. Die Entwicklerszene scheint auf eine solche Innovation nur gewartet zu haben, denn während Commodore im Sommer '88 erst einmal den 68020-AMIGA vorstellen wird und Informationshäppchen über einen 68030-AMIGA eindeutig in der Gerüchteküche zubereitet wurden, sind Transputererweiterungen für den AMIGA bereits im Prototypen-Stadium. Aus diesem Grund möchten wir Ihnen in diesem Artikel etwas Einlick in die Grundlagen und Konzepte, die den neuen Prozessoren zugrunde liegen, vermitteln.

Der 68030...

Mit dem 68030 stellt Motorola den Nachfolger des 68020 und damit in gewisser Weise auch einen Nachkommen des im AMIGA verwandten 68000 vor. Die Neuerung bei diesem Chip liegt neben der schon obligatorischen Geschwindigkeitssteigerung in der auf dem Chip integrierten Speicherverwaltungseinheit. Entgegen vielen anderen Chipneuheiten benötigt der 68030 keine neue Software. Durch das aufwärtskompatible Codeformat wird es ermöglicht, bestehende 68000/68020-Software relativ problemlos zu übernehmen. Da bereits heute über auf dem 68030 basierende AMIGA-Modelle spekuliert wird (während die AMIGA-Fangemeinde noch gespannt die Vorstellung des ersten 68020-AMIGA erwartet), soll hier schon einmal vorab das interessante Konzept des 68030 beschrieben werden.

...und der T800

Schon ein einzelner dieser T800-Chips bringt eine enorme Rechenleistung, die mit der eines 68030 zusammen (!) mit Arithmetikcoprozessor 68882 vergleichbar ist. Doch seine wahre Stärke zeigt der Chip, der zur Gruppe der Transputer gehört, erst in der Gemeinschaft seinesgleichen. Hier wächst die Rechengeschwindigkeit des Systems nahezu linear mit der Anzahl der Prozessoren an. Nachdem der Transputer in letzter Zeit in aller Munde ist, wird nun auch bei Commodore an der Entwicklung von Transputerkarten für den AMIGA gearbeitet, während ein deutscher Hersteller, die Firma Sang Computersysteme, der KICKSTART-Redaktion auf der CeBIT 88 bereits einen Prototypen einer Transputerkarte für den AMIGA 2000 präsentieren konnte, die in naher Zukunft einsatzfähig und für einen Test verfügbar sein soll.

Das Ziel lautet Geschwindigkeit!

Eine Methode zur Geschwindigkeitssteigerung nutzen beide Prozessoren in hohem Maße aus: Die interne Parallelverarbeitung. Alle Einheiten auf den Prozessorchips arbeiten weitgehend unabhängig. Nur bei Bedarf kommunizieren sie miteinander. Um dabei ein gegenseitiges Warten möglichst auszuschließen, wurden zwischen den Funktionsblöcken, wo es sinnvoll ist, Pipelines angelegt. Pipelines sind Zwischenspeicher nach dem FIFO (First In - First Out)-Prinzip, das ähnlich einer Warteschlange im Supermarkt funktioniert. Was man mit solchen Parallelisierungsmaßnahmen erreichen kann, sollen folgende Beispiele zeigen: Der 68030 kann einen Befehl, der nur Daten im Prozessor ändert, gleichzeitig mit einem Befehl, der externe Daten ändert, ausführen. Außerdem kann die auf dem Chip vorhandene Speicherverwaltungseinheit Adressen berechnen, während andere Einheiten Befehle bearbeiten.

Beim T800-Transputer können ebenfalls zwei Befehle simultan bearbeitet werden, einmal vom Prozessor und einmal von der auf dem Chip befindlichen Arithmetikeinheit. Während komplizierterer Realberechnungen kann die Zeit für andere Aufgaben genutzt werden. Besonders hohe Geschwindigkeit verleihen diesem Chip die ebenfalls parallel arbeitenden seriellen Schnittstellencontroller, die über DMA verfügen, und die ebenfalls, parallel zu allem bisher Geschilderten, mögliche Blockkopierfunktionen im internen Speicher unterstützen.

Das alte Problem: RISC, CISC und der Flaschenhals

Als RISC (Reduced Instruction Set Computer) werden Systeme bezeichnet, deren Zentraleinheit (CPU) mit nur wenigen Befehlen auskommt. Der Mangel an Befehlen wird durch eine große Anzahl an Registern ausgeglichen. Der Vorteil dieses Konzepts ist die hohe Bearbeitungsgeschwindigkeit der Befehle. Meist wird pro Taktzyklus ein Befehl durch die Maschine geschaufelt. Außerdem vereinfacht eine kleine Befehlsanzahl die Entwicklung hochoptimierender Compiler - die direkte Programmierung in Maschinensprache allerdings wird grauselig. Ohne Zusatzprogramme hat man fast keine Chance, einen richtigen RISC-Prozessor optimal zu programmieren. Ein reiner Vertreter des RISC-Prinzips ist der ARM-Chip (Acorn Risc Machine) des Acorn Archimedes, der bei gleicher Taktfrequenz durchschnittlich 4mal schneller als ein 68000er arbeitet.

Die Transputerchips, welche in der

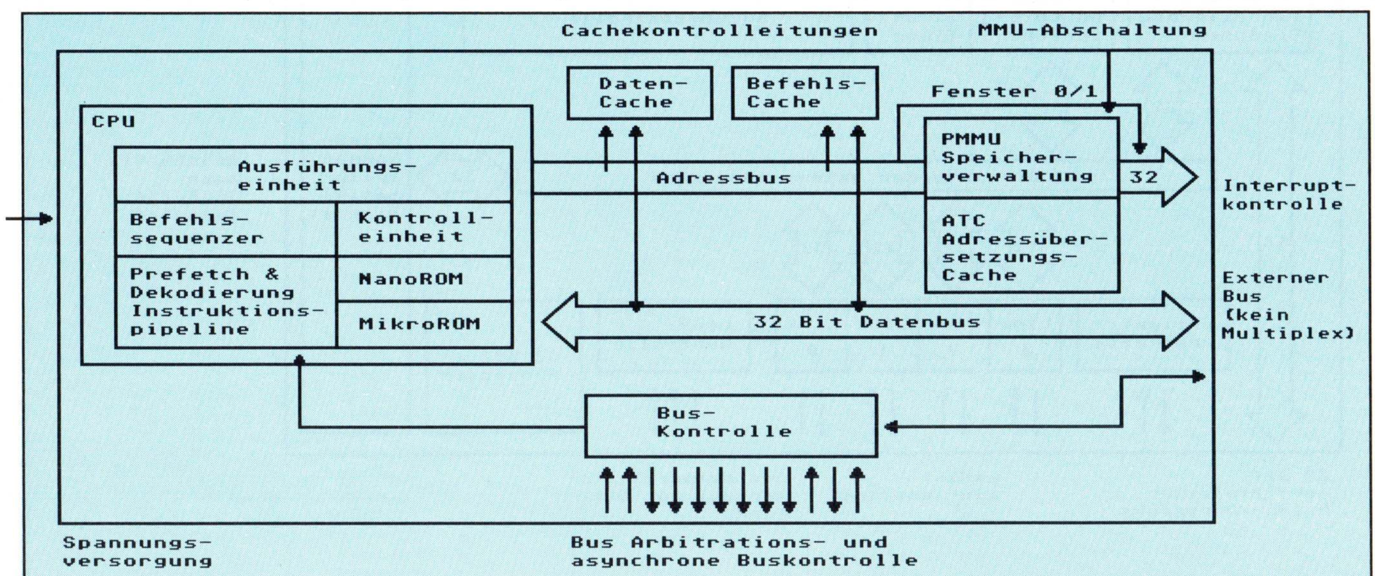


Bild 1: Die 68030 Prozessorarchitektur

Transputerkarte von Sang und irgendwann auch in einer Commodore-Transputerkarte Verwendung finden, sind Zwitterwesen, die man am ehesten als RISC-Chips bezeichnen kann. Ihr Konzept weicht allerdings etwas von der RISC-Philosophie ab: Die Registeranzahl ist beschränkt. Dieser Mangel an Registern wird allerdings durch den großen Speicher auf dem Prozessor mehr als ausgeglichen, denn dieser statische Speicher hat durch seine hohe Geschwindigkeit Registereigenschaften. Außerdem ist entgegen der RISC-Philosophie ein relativ großer Befehlsvorrat vorhanden. Hier wurden durch einen Trick Geschwindigkeitsverluste verhindert (dazu später mehr).

Der 68030 entspricht mehr dem CISC-Konzept (Complex Instruction Set Computer - Komplexer Befehlssatz), welches sich durch eine große Anzahl verschiedenartiger Befehle mit sehr vielen Adressierungsarten (Kodierungsformen für Speicherzugriffe) auszeichnet. CISC-Prozessoren sind daher recht einfach auch per Hand optimal zu programmieren. Ihr Befehlssatz kommt einer Hochsprache wesentlich näher als der von RISC-Systemen. Durch seine relativ große Registeranzahl besitzt er auch RISC-Merkmale. Ein weiteres Merkmal für RISC und CISC ist die Ausführungszeit der Befehle. Bei RISC-Systemen wird das Ziel verfolgt, alle Befehle möglichst in einem Taktzyklus abzuarbeiten. Bei CISC-Systemen wird dies meist durch den hohen Dekodierungsaufwand für die vielen verschiedenartigen Befehle unmöglich. Um trotzdem hohe Geschwindigkeiten zu erzielen, versucht man zeitaufwendige Befehle möglichst komplett durch optimierte Hardware abzuarbeiten.

Es stellt sich allerdings die Frage, was an dem 68030 gegenüber seinen Vorgängern, dem 68020 und dem Ihnen bekannten 68000 des AMIGA, das Neue ist. Wenn man noch einmal RISC- und CISC-Chips vergleicht, stellt man fest, daß es hier meist nur scheinbare Vorteile gibt. Zwar lassen sich RISC-Chips normalerweise leichter entwickeln und produzieren, doch sind auch CISC-CPUs mit hoher Leistung und vereinfachter Entwicklung denkbar (z.B. der 68030). Der Trend geht jedoch eindeutig zu schnellen RISC-Prozessoren, wobei allerdings versucht wird, einige der CISC-Vorteile zu übernehmen.

Doch bei allen Optimierungsmaßnahmen, die die Bauelemente teilweise enorm aufwendig und auch teuer machen, hat man Überlegungen angestellt, wie anderweitig die Leistung gesteigert werden kann. Es hat sich herausgestellt, daß eines der schwächsten Glieder der Kette bei allen Computern der externe Speicher ist (daher auch die vielen schnellen Register bei RISC-Chips). Alle herkömmlichen Prozessoren - also auch Transputer - bearbeiten einen Befehl nach dem anderen. Dazwischen wird der jeweils nächste Befehl aus dem Speicher geladen. Dies geschieht über den gleichen Bus wie die Datenschreib- und Lesezugriffe. Programmbefehle und Daten liegen im gleichen Speicher. In der Vergan-

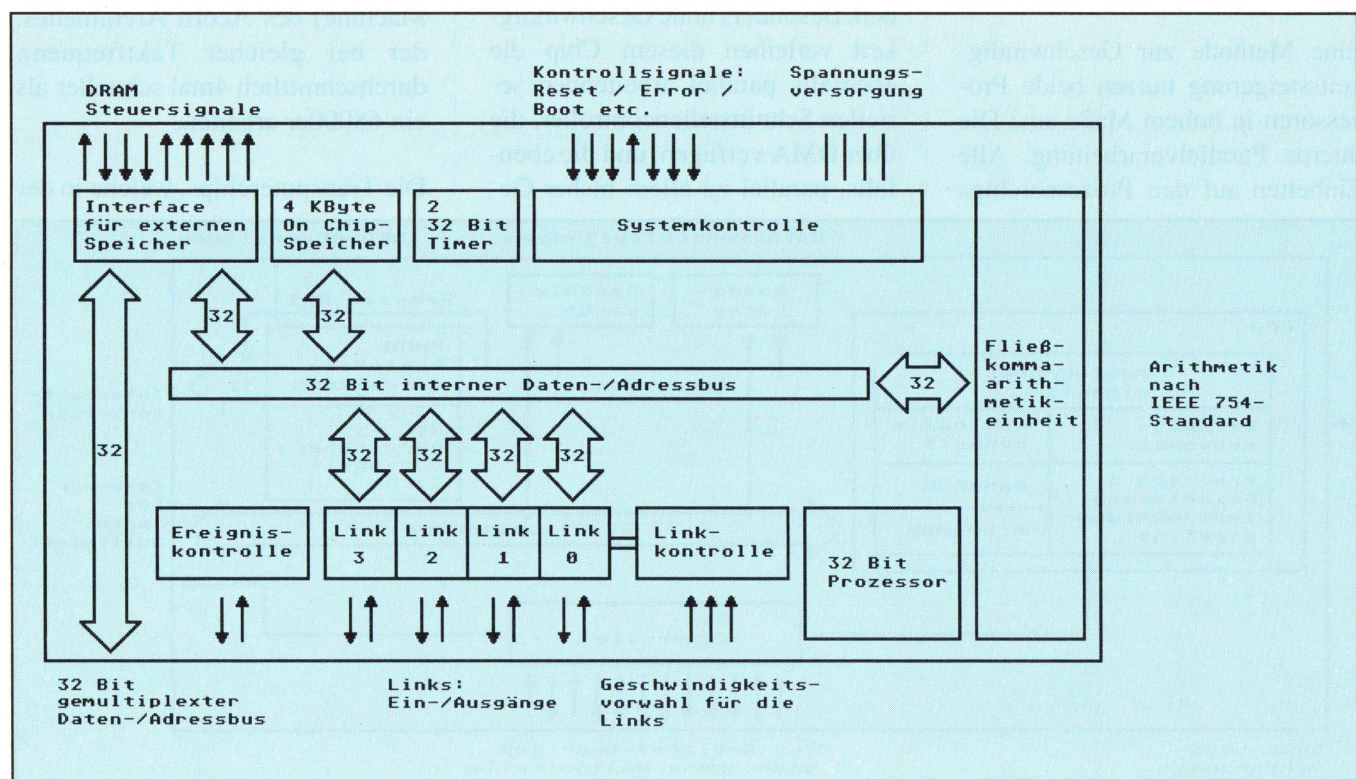


Bild 2: Die IMS T800 Transputerarchitektur

genheit hat es viele Versuche gegeben, diesen Flaschenhals zumindest zu erweitern. Am bekanntesten dürfte die Methode des CPU-Cache-Speichers sein. Hierbei wird der CPU ein extrem schneller Speicher zugeordnet. Programmschleifen können jetzt mit maximaler Geschwindigkeit bearbeitet werden. Doch alle diese Verfahren beseitigen nicht das eigentliche Problem: den nach dem Erfinder der bisherigen Prozessorarchitektur benannten von Neumann-Flaschenhals...

Lösung Nr. 1: Die Harvard-Architektur

Sowohl der 68030 als auch Motorolas neuer RISC-Prozessor 88000 (für Freaks: Leistung min. 17 MIPS) verwenden diese Architektur in teilweise modifizierter Form. Durch einen getrennten Daten- und Instruktionsbus wird die Leistungsfähigkeit gesteigert. Im Idealfall kann so zeitlich parallel ein Befehl gelesen werden, während ein noch in der Ausführung befindlicher Befehl gerade Daten auf den Datenbus schreibt. Bei einer entsprechenden Parallelisierung der Prozessorarchitektur kann so der Durchsatz enorm gesteigert werden. Um den Aufwand bei der externen Hardware nicht zu vergrößern, hat man das parallele Bussystem beim 68030 nur im Prozessor realisiert. Ein Cachespeicher für jeden Bus sorgt dafür, daß die Harvard-Architektur ihren Sinn nicht verliert. Beim 68030 sind die Cache-Speicher jeweils 256 Byte groß und extrem schnell.

Lösung Nr. 2: Das Transputer-Konzept

Eine endgültige Lösung des Problems stellt ein paralleles Computerkonzept dar. Inmos' Transputerchips mit ihren Kommunikationskanälen sind ein Beispiel für eine Computerarchitektur mit Parallelität. Hierbei kann auch der schnellste Cache oder Bus nicht mehr mithalten: Jeder sogenannte Transputer besitzt auf dem Chip 4 sehr schnelle serielle Schnittstellen, die Links. Beim T800 können diese Links mit bis zu 20 MBit/s Daten übertragen. Dies bedeutet bei bidirektionalem Betrieb, daß weit mehr als 2 MegaByte pro Sekunde übertragen werden können. Durch die asynchrone Betriebsart der Links ist es noch nicht einmal erforderlich, daß außer über die Links irgendeine Beziehung zwischen den Kommunikationspartnern besteht. Beliebig viele Prozessoren können somit zu einem riesigen Netz verschaltet werden.

Untersuchungen haben nun ergeben, daß unterschiedliche Netzstrukturen für bestimmte Anwendungen Vorteile haben. Sequentielle Probleme lassen sich auf einer Kette von Transputern besonders schnell bewältigen (Prinzip der Serienproduktion). Für komplexe Schleifen kann man einen Ring

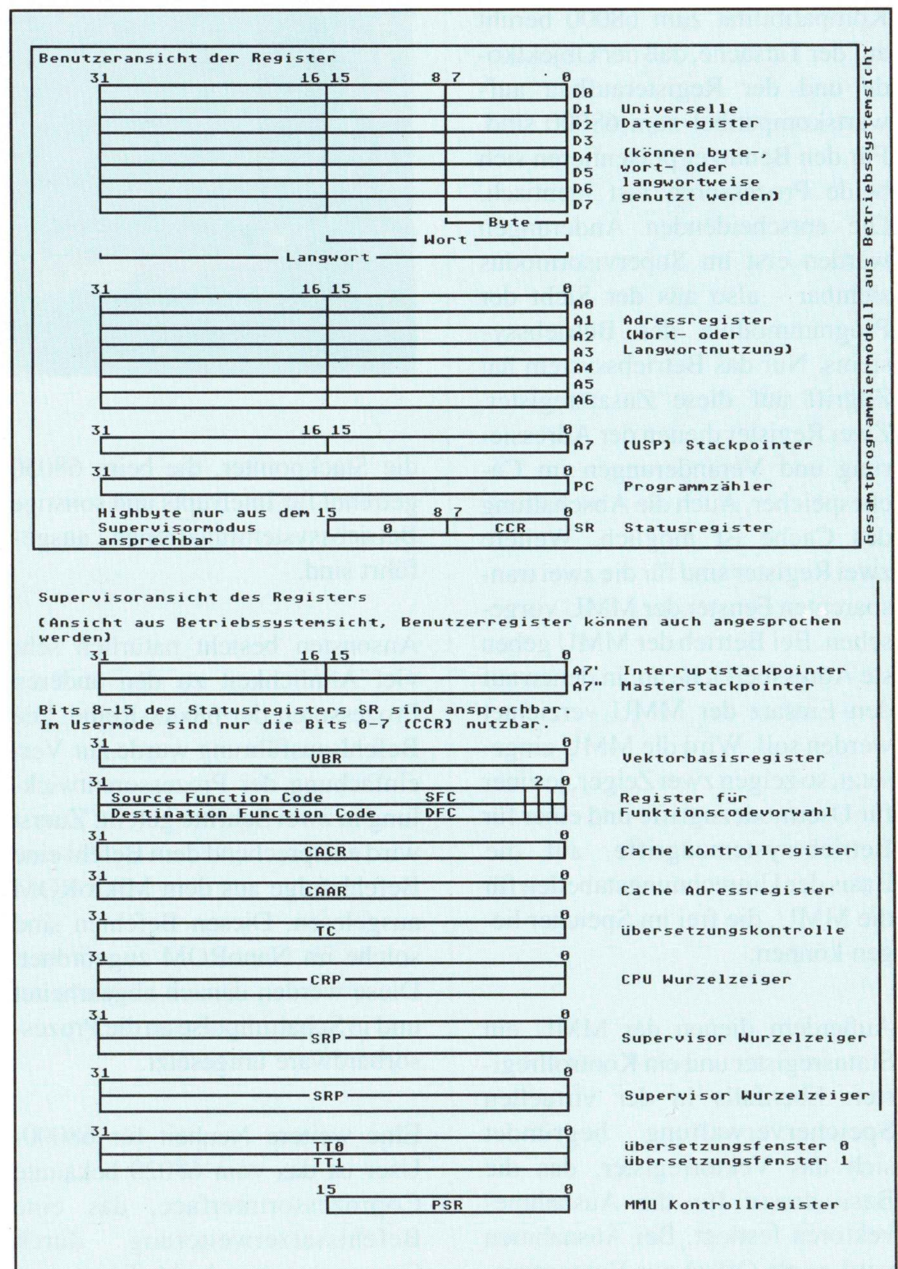


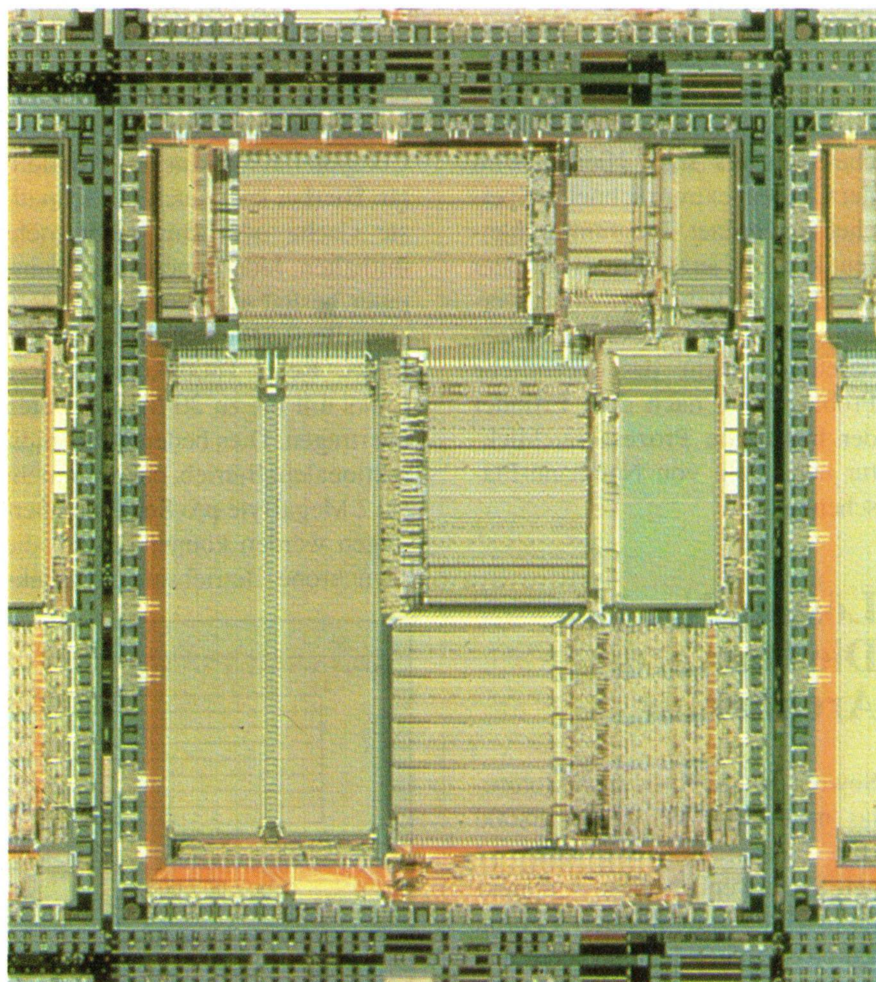
Bild 3: Das Registermodell des 68030

zusammenschalten, und wieder andere Probleme sind optimal auf einer dreidimensionalen Struktur zu lösen.

Der 68030 im Einzelnen: Weitere Unterschiede zum 68000

Da die meisten nicht über weitgehende Erfahrungen bei der Programmierung des 68020 verfügen, soll hier ein Vergleich zum 68000 gezogen werden. Die weitgehende Kompatibilität zum 68000 beruht auf der Tatsache, daß der Objektkode und der Registeraufbau aufwärtskompatibel zum 68000 sind. Für den Benutzer präsentieren sich beide Prozessoren fast identisch. Die entscheidenden Änderungen werden erst im Supervisormodus sichtbar - also aus der Sicht der Programmodule des Betriebssystems. Nur das Betriebssystem hat Zugriff auf diese Zusatzregister. Zwei Register dienen der Adressierung und Veränderungen im Cachespeicher. Auch die Abschaltung des Cache ist möglich. Weitere zwei Register sind für die zwei transparenten Fenster der MMU vorgesehen. Bei Betrieb der MMU geben sie Adressbereiche an, in denen auf den Einsatz der MMU verzichtet werden soll. Wird die MMU eingesetzt, so zeigen zwei Zeiger, je einer für Usermodezugriffe und einer für Betriebssystemzugriffe, auf die Basis der Umrechnungstabellen für die MMU, die frei im Speicher liegen können.

Außerdem dienen der MMU ein Statusregister und ein Kontrollregister. Ebenfalls in der virtuellen Speicherverwaltung begründet sich das Vektorregister, das die Basisadresse für die Ausnahmevektoren festlegt. Bei Ausnahmen wird es als Offset zur Vektornummer addiert. Was noch fehlt, sind die Stackpointer, die beim 68030 getrennt für Interrupts und sonstige Betriebssystemfunktionen ausgeführt sind.



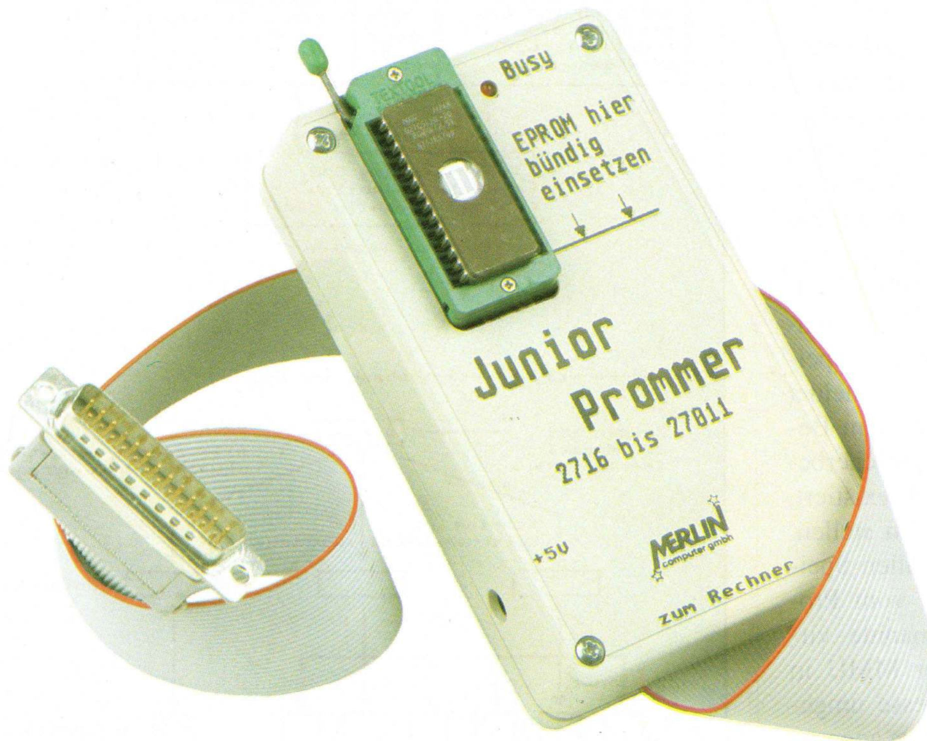
Ansonsten besteht natürlich sehr viel Ähnlichkeit zu den anderen Prozessoren der 68xxx-Reihe. Die Befehlsausführung wurde zur Vereinfachung der Prozessorentwicklung in zwei Schritte geteilt: Zuerst wird entsprechend dem Befehl eine Befehlsfolge aus dem MikroROM ausgelesen. Diesen Befehlen sind solche im NanoROM zugeordnet. Diese werden danach abgearbeitet und in Schaltimpulse an die Prozessorhardware umgesetzt.

Eine weitere Neuheit für 68000-User ist das vom 68020 bekannte Coprozessorinterface, das eine Befehlssatzerweiterung durch Coprozessoren erlaubt. Treten nun Befehle für diese Coprozessoren auf, so übernimmt der Prozessor die Adressberechnung und den Datentransfer. Die eigentliche Berechnung (z.B. Realarithmetik im 68881/68882) findet parallel im Coprozessor statt.

Eine MMU on chip

Was für den 68020 als Coprozessor erhältlich ist, nämlich eine MMU, wurde beim 68030 gleich integriert, denn alle komplexeren Betriebssysteme, insbesondere UNIX, sind vom Vorhandensein eines solchen Bauelementes abhängig. Was auf den 8-Bit-Computern die statische Speicherverwaltung war und auf den 16-Bit-Maschinen die dynamische Speicherverwaltung ist, das ist bei 32-Bit-Prozessoren und entsprechenden Betriebssystemen die MMU. Eine MMU (Memory Management

Klein, kompakt und leistungsstark- der JuniorPrommer



Der JuniorPrommer programmiert alle gängigen EPROM-Typen, angefangen vom 2716 (2 KByte) bis zum modernen 27011 (1 MBit). Aber nicht nur EPROMS, sondern auch einige ROM- und EEPROM-Typen lassen sich lesen bzw. programmieren.

Zum Betrieb benötigt der JuniorPrommer nur +5V, die am Joystickport Ihres Amiga abgenommen werden; alle anderen Spannungen erzeugt die Elektronik des JuniorPrommers. Die sehr komfortable Software, natürlich Menü-unterstützt, erlaubt alle nur denkbaren-Manipulationen.

Fünf Programmieralgorithmen sorgen bei jedem EPROM-Typ für hohe Datensicherheit. Im eingebauten Hex-/ASCII-Monitor läßt sich der Inhalt eines EPROMS blitzschnell durchsuchen und auch ändern.

Alles dabei!
Bemerkenswert ist der Lieferumfang, so wird z.B. das Fertiggerät komplett aufgebaut und geprüft im Gehäuse mit allen Kabeln anschlußfertig geliefert.

Auf der Diskette mit der Treibersoftware befindet sich noch ein Programm, das die Kickstart-Diskette in vier EPROM-Dateien für 27512 (64 KByte EPROM) zerlegt, die dann mit dem JuniorPrommer gebrannt werden können. Ferner wird der Source-Code für die Lese- bzw. Programmierroutinen mitgeliefert. Und last but not least ist im Bedienungshandbuch (deutsch) der Schaltplan abgedruckt.



Bestellcoupon Merlin Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Hiermit bestelle ich:

- ☐ Fertiggerät wie oben beschrieben DM 249,00
- ☐ Leerplatine (o. Bauteile) und Software DM 59,00
- ☐ Leergehäuse (gebohrt und bedruckt) DM 39,90

Versandkosten: Inland DM 7,50
Ausland DM 10,00
Auslandsbestellungen nur gegen Vorauskasse
Nachnahme zuzüglich DM 3,50 Nachnahmegebühr

Unit) verwaltet den Speicher. Sie sorgt für größere Portabilität der Programme. Bei Betriebssystemen, die mehrere Benutzer und gleichzeitig laufende Programme (Multitasking) erlauben, muß gewährleistet sein, daß sich Programme nicht gegenseitig stören. Außerdem hilft die MMU, falls ein Programm größer ist als der Speicher. Auf dem 68030 lassen sich Programme schreiben, die grundsätzlich von 4 GigaByte vorhandenem Hauptspeicher ausgehen. Wird nun vom Programm ein entsprechender, nicht real vorhandener Speicherbereich adressiert, so meldet die MMU dies und hilft dem Betriebssystem, diesen Speicherbereich, der ja nicht real existiert, logisch zur Verfügung zu stellen. Dies geschieht so: Bei allen Speicherzugriffen wird die vom Programm erzeugte sogenannte virtuelle Adresse von der MMU in eine reale Speicheradresse umgesetzt. Eine vom Programm angegebene logische Adresse 0 kann also durch die MMU an fast beliebiger Stelle im Speicher liegen.

Um das Chaos nicht komplett zu machen, geschieht dies seitenweise. Das heißt, für einen bestimmten Bereich im physikalischen (realen) Speicher wird eine logische Basisadresse angegeben. Bei deren Auftreten wird nun die entsprechende Speicherseite verwendet. Jeweils 22 logische Basisadressen werden in einem Assoziativspeicher gehalten, der eine Adressumsetzung ohne Zeitverlust ermöglicht.

Falls ein bestimmtes Speichersegment nicht im Assoziativspeicher der MMU gefunden wird, so wird eine Ausnahme ausgeführt, und die betreffende Seite kann durch das Betriebssystem von einem Massenspeicher geladen werden. Um die Anzahl der Seiten nicht auf 22 zu beschränken, können beliebige Umrechnungstabellen, die eine Baumstruktur bilden, angelegt werden. Diese Baumstruktur bietet

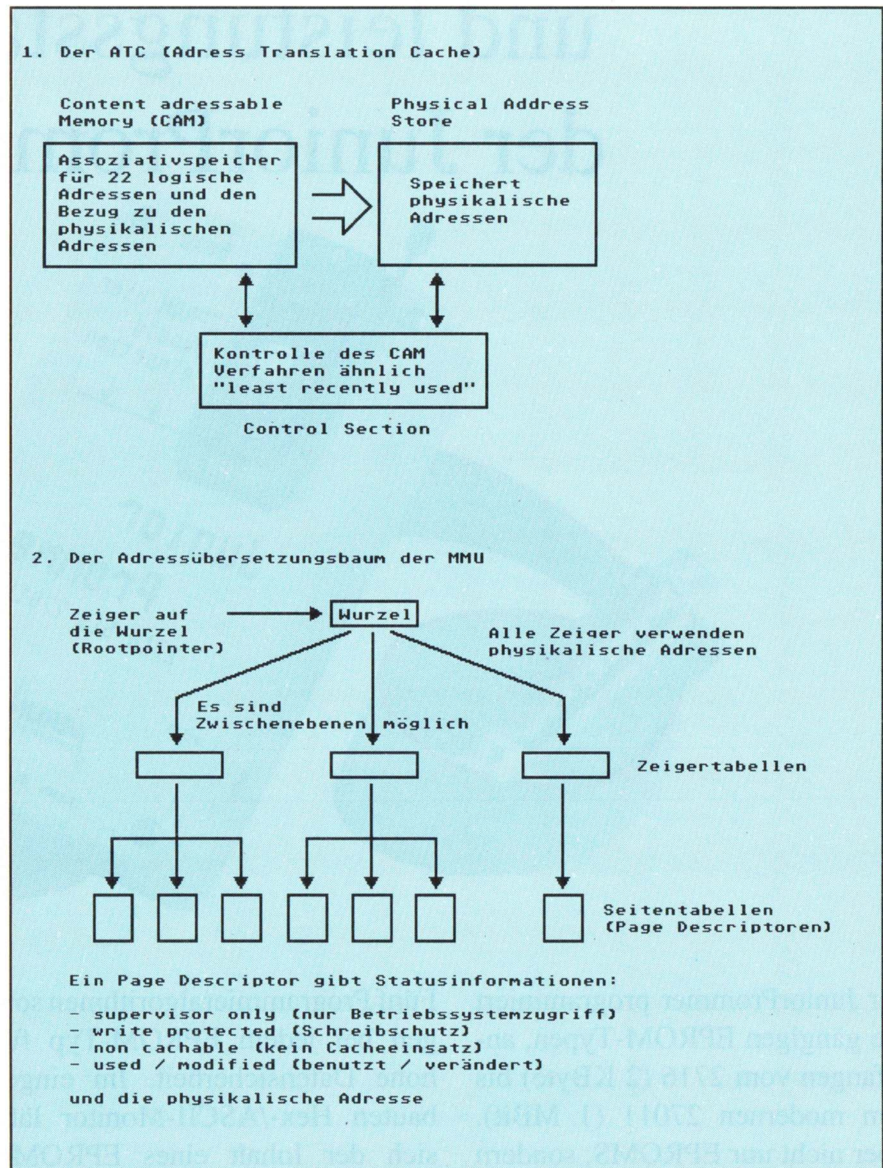


Bild 4: Die Paged Memory Managment Unit (PMMU) des 68030

neben hoher Geschwindigkeit ein Höchstmaß an Flexibilität. Eine ganze Gruppe von Seiten kann durch Veränderung einer höheren Hierarchieebene in ihrer Zugriffsmöglichkeit verändert werden. In den die Speicherseiten beschreibenden Kontrollworten kann markiert werden, ob eine bestimmte Speicherseite nur vom Betriebssystem verwandt werden darf, diese Seite nur gelesen werden darf, oder bei Zugriffen auf diese Seite der Cache nicht benutzt werden soll. Dies ist eine große Hilfe bei größeren Systemen mit eventuell mehreren unabhängigen Benutzern.

Zur Kompatibilität mit seinen kleineren Brüdern kann beim 68030 die Speicheradressumrechnung abgeschaltet werden, und für Zugriffe zum Beispiel auf im Speicher liegende Peripheriebausteine können zwei Transparentfenster eingerichtet werden, in denen keine Adressumrechnung stattfindet.

Zusammenfassend

Beim 68030 wurden auf interessante Weise die Vorteile des RISC- und des CISC-Konzeptes kombiniert. Die aufwendigen Optimierungen gegenüber dem 68020 und vor allem dem 68000, die unter anderem auf der Harvard-Architektur und der integrierten MMU basieren, machen den 68030 zu einem

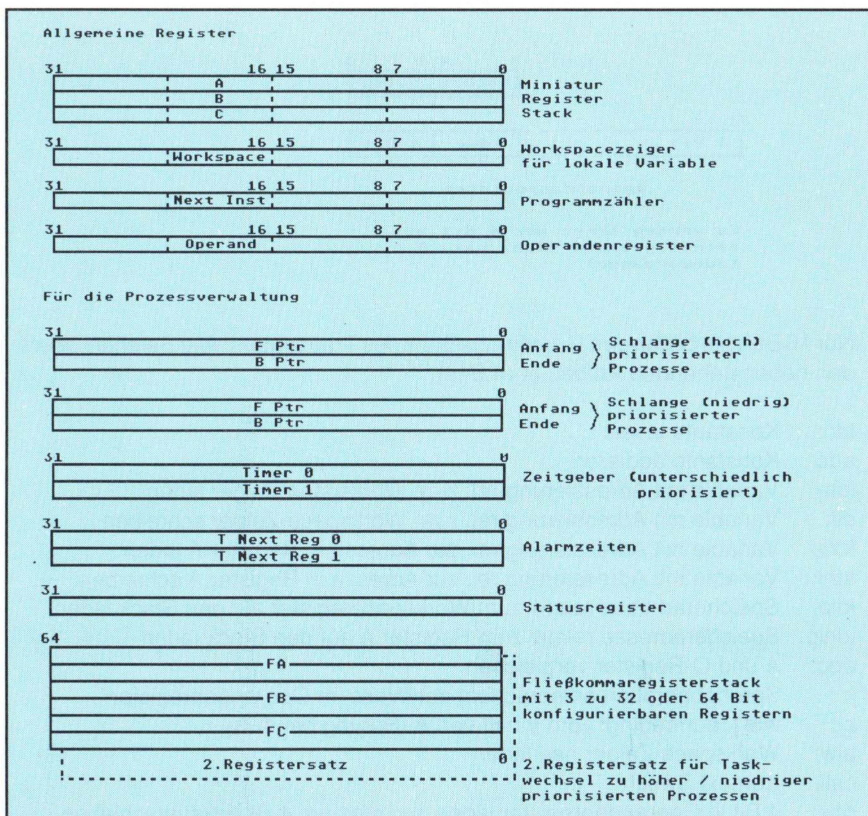


Bild 5a: Das Registermodell des T800

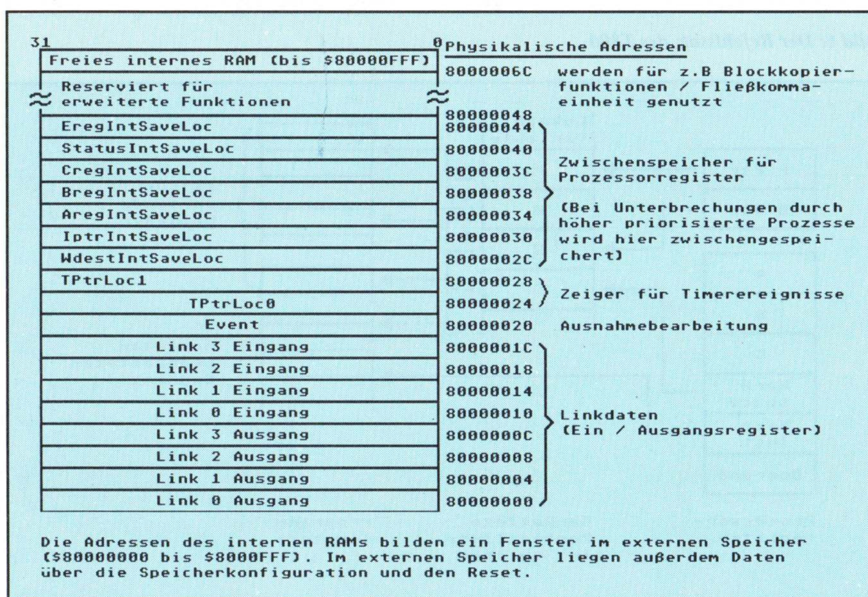


Bild 5b: Die Speicherbelegung des internen 4kByte RAMs

extrem leistungsfähigen Prozessor. Die weitgehende Softwarekompatibilität zu seinen Vorgängern zusammen mit bis zu 5 MIPS (Millionen Befehle pro Sekunde) machen ihn zu einem idealen Prozessor für Lowcost-Workstations der Zukunft. Wem die Leistung nicht reicht, der kann durch den Einsatz des 68882-Aritmetikprozessors, der mit dem 68030 als Coprozessor zusammenarbeitet, die Leistung seines Systems noch weiter steigern. Übrigens: der 68882 ist der pinkompatible Nachfolger des 68881. Optimierungen durch Motorola haben den Durchsatz bei gleicher Taktfrequenz auf das 1,5-fache steigern können.

Der T800 im einzelnen: Transputer-Grundlagen

Was heißt das eigentlich "Transputer" bzw. was kann man sich darunter vorstellen? Zunächst einmal ist dies ein Mikroprozessor wie auch der 68000, doch unterscheidet er sich von diesem durch einige gravierende Dinge.

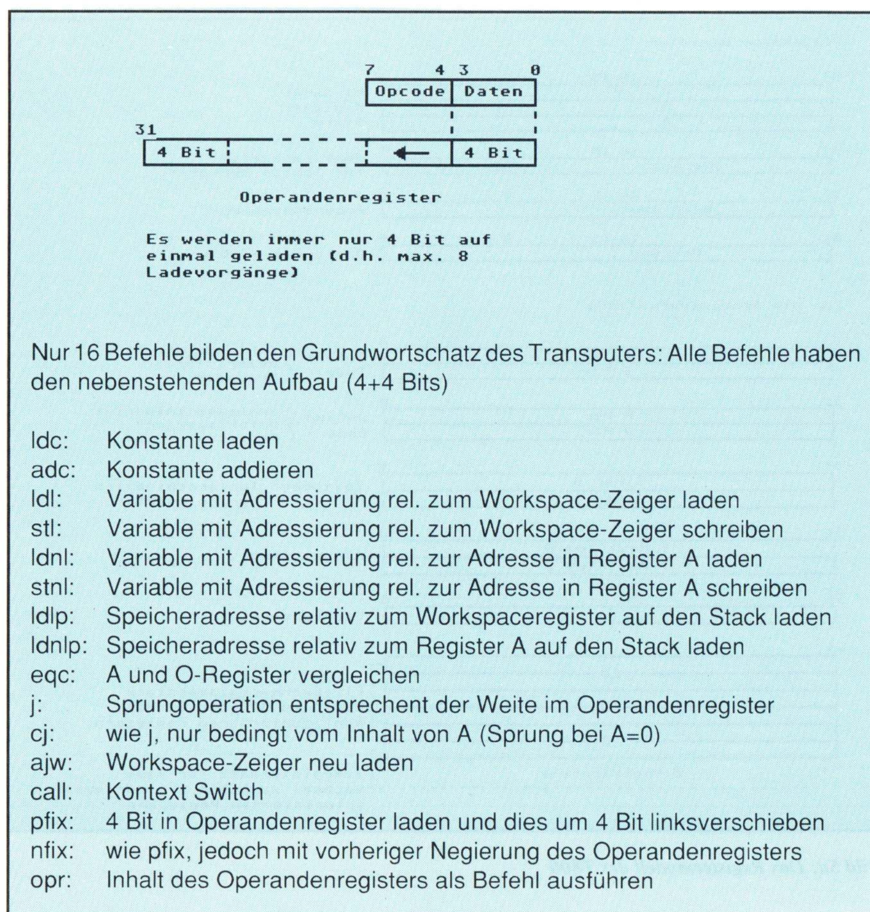
Der eine Unterschied ist sein Befehlssatz. Während der 68000 mit vielen Befehlen und fast ebensovielen Adressierungsarten glänzt, ist der Grundbefehlsvorrat des Tranputer eher bescheiden. Auch bei der Registeranzahl schneidet der Transputer schlechter ab. Als Arbeitsregister dienen nur 3 Register, die in Stackmanier verwaltet werden. Doch was ist der Vorteil dieser Bescheidenheit? Ganz einfach: Einmal abgesehen von der vereinfachten Entwicklungsarbeit für das Chiplayout, bringt die Reduzierung eine Geschwindigkeitssteigerung. Die einfachen Instruktionen benötigen zu ihrer Kodierung weniger Speicherplatz (und können damit schneller in den Prozessor geladen werden), und da es nicht sehr viele Befehle gibt, kann man sie hardwaremäßig in das Silizium bannen. Beim 68000 werden die Befehle nicht direkt per Hardware ausgeführt. Hier befindet sich eine Art kleiner Computer im Prozessor. Da die Entwickler auch bei diesem Chip Zeit beim Chipdesign sparen wollten, wurden die Befehle softwaremäßig implementiert. Ein realer Befehl führt zur Ausführung eines Programms aus dem Mikro- bzw. NanoROM im Prozessor. So ist es auch möglich, den 68000 mit individuellem Befehlssatz zu beziehen. Wenn Sie die "wenigen" Dollars für die Maskenherstellung usw. aufbringen können, ist man bei Motorola sicher gern bereit, Ihnen einen solchen Chip zu liefern.

Doch zurück zum Befehlssatz des Transputers. Wie sich bei statistischen Analysen von Programmen gezeigt hat, werden bestimmte Befehle häufiger und andere seltener benutzt. Beim Transputerbefehlssatz wurde dies natürlich berücksichtigt. Die meistbenutzten Befehle sind durch ein Byte kodierbar, komplexere und seltenere Befehle werden durch bestimmte Aneinanderreihungen der einfachen Instruktionen ausgelöst.

Zusatzfunktionen in Hardware

Neben der normalen Befehlsbearbeitung kann der Transputer noch einiges mehr. Beispielsweise unterstützt er Multitasking bereits hardwaremäßig. Das Programmodul, welches normalerweise für den Taskwechsel sorgt, der Scheduler, wurde einfach hardwaremäßig integriert. Eine verkettete Liste enthält Zeiger auf ausführbare Programmsegmente. Zwei Prozessorregister zeigen auf das Ende und den Anfang dieser Liste. Im Beispiel warten die Prozesse P, Q, und R auf Ausführung, S wird momentan ausgeführt. Falls ein Prozeß auf eine Ein- oder Ausgabe über Links wartet o.ä., reserviert der Scheduler automatisch keine Zeit mehr für ihn. Für den Scheduler ebenfalls notwendig sind Timer. Davon gibt es wie von den Prozeßlisten zwei: einen für höher und einen für niedriger priorisierte Prozesse.

Interessant beim Multitaskingkonzept innerhalb der Transputer ist die Tatsache, daß Register - auch die der Floating-Point-Einheit - bei einem Wechsel von einem niedriger zu einer höhem priorisierten Task automatisch gerettet werden. An weiteren Sonderfunktionen bietet der Transputer eine CRC-Prüfsummenfunktion und Unterstützung für Kopiervorgänge zweidimensionaler Felder, z.B. für Grafiken.



Nur 16 Befehle bilden den Grundwortschatz des Transputers: Alle Befehle haben den nebenstehenden Aufbau (4+4 Bits)

ldc:	Konstante laden
adc:	Konstante addieren
ldl:	Variable mit Adressierung rel. zum Workspace-Zeiger laden
stl:	Variable mit Adressierung rel. zum Workspace-Zeiger schreiben
ldnl:	Variable mit Adressierung rel. zur Adresse in Register A laden
stnl:	Variable mit Adressierung rel. zur Adresse in Register A schreiben
ldlp:	Speicheradresse relativ zum Workspaceregister auf den Stack laden
ldnlp:	Speicheradresse relativ zum Register A auf den Stack laden
eqc:	A und O-Register vergleichen
j:	Sprungoperation entsprechent der Weite im Operandenregister
cj:	wie j, nur bedingt vom Inhalt von A (Sprung bei A=0)
ajw:	Workspace-Zeiger neu laden
call:	Kontext Switch
pfix:	4 Bit in Operandenregister laden und dies um 4 Bit linksverschieben
nfix:	wie pfix, jedoch mit vorheriger Negierung des Operandenregisters
opr:	Inhalt des Operandenregisters als Befehl ausführen

Bild 6: Der Befehlssatz des T800

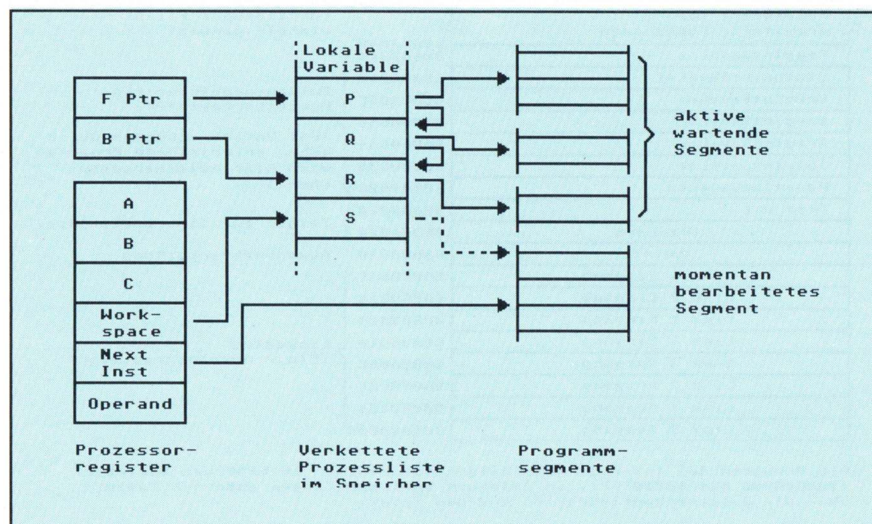


Bild 7: Der Hardware-Scheduler im Transputer

T800 - Der Floating-Point-Transputer

Den Vogel unter allen Transputerchips schießt eindeutig der T800 ab. Eine Kombination aus einem superschnellen 32 Bit-Mikroprozessor und einer ebensoschnellen Arithmetikeinheit für Fließkommaarithmetik bildet den optimalen Baustein für ein Transputernetzwerk. Die Firma Sang verwendet auf ihrer Transputerkarte für den AMIGA 2000 die 20 MHz-Version. Diese 20 MHz bringen den Prozessor gehörig auf Trab. Da auch die Arithmetikeinheit parallel zu allen anderen Einheiten des Prozessors arbeitet, kann zusätz-

lich zu den mit 20 MHz durch die CPU geschaufelten Befehlen (ca. 10 MIPS = 10 Mio. Befehle/sec) eine Rechenleistung von ca. 1,5 MFLOPS (= Mio. Fließkommaoperationen/sec) genutzt werden. Die Fließkommaoperationen entsprechen dem IEEE 754-Standard und damit dem Format der meisten Compiler und Programme. 1,5 MFLOPS werden bei Operationen mit einfacher Genauigkeit erreicht (32 Bit = 7 Stellen), bei doppelter Genauigkeit (64 Bit) wird eine höhere Genauigkeit (15 Stellen) mit einer verringerten Geschwindigkeit bei Multiplikations- und Divisionsbefehlen erreicht. Die sonstigen Befehle (z.B. Addition) werden durch die Genauigkeitserhöhung nicht beeinflußt. Was diese Geschwindigkeit bedeutet, soll ein Vergleich mit dem AMIGA zeigen: Der 68000 schafft ca. 25.000 Multiplikationen oder max. 60.000 Additionen im Fließkommaformat (einfache Genauigkeit!). Dies bedeutet einen Geschwindigkeitsvorteil von ca. 60-120.

Fazit

Mit den neuen Prozessoren ist der technologische Standard von Mikroprozessoren um einen großen Schritt vorangebracht worden. Obwohl die Preise der neuen Chips vorerst keine neuen Heimcomputer erwarten lassen, liegen gerade bei Transputerkarten die von den Herstellern angestrebten Preise in Relation zur Leistungsfähigkeit dieser Systeme auf einem bisher unerreichten Niveau. Ein 68030-AMIGA ist, wie bereits erwähnt, noch in den Bereich der Spekulation einzuordnen, während Transputerkarten für den Amiga bereits in naher Zukunft erhältlich sein werden. Hier stellt sich zwar noch die Frage nach der Unterstützung einer solchen Erweiterung durch das Betriebssystem, die sich für den Anfang sicherlich in engen Grenzen halten wird, was aber für professionelle Anwender (für die solche Karten mit ihrer hohen Rechenleistung und einem Preis, der sich nicht gerade im Heimcomputerpreisbereich bewegt, konzipiert sind) erst einmal von geringerem Interesse sein wird.

Rechenintensiveren Aufgaben im Bereich der Simulation und wissenschaftlichen Anforderungen ist ein solches Transputer-System auf Basis des T800 sicherlich gewachsen. Schon ein einzelner Chip läßt einen VAX-Rechner weit hinter sich. Viele Berechnungen werden erst mit Transputersystemen zu vernünftigen Preisen realisierbar. Eine faszinierende Anwendung dürfte auch die Erstellung hochwertiger Grafiken mit dem System sein (Ray Tracing mit 1.5 MFLOPS, der Traum eines Sculptgeplagten Grafikfreunds!). Man darf also insgesamt auf die weiteren Entwicklungen sehr gespannt sein. Die KICKSTART-Redaktion wird sich auf jeden Fall bemühen, das Transputersystem von Sang zu testen, sobald es zur Serienreife gelangt ist.

ENDE

Kurzdaten des 68030:

* Quell- und Objektcodekompatibel zum 68020 und 68000
* 32-Bit Adress- und 32-Bit Daten-Bus (4 GByte Adressraum)
* 16 32-Bit Universalregister (wie 68000)
* 2 Supervisorstackpointer und 10 Spezialregister für Steueraufgaben
* 256 Byte Cache für Daten und 256 Byte Cache für Programmkode
* MMU für seitenw. Speicherverwaltung (Seitengröße: 256 - 32KB)
* 2 Transparentsegmente per MMU wählbar
* Pipeline-Architektur mit interner Parallelverarbeitung
* 18 Adressierungsarten und 7 Datentypen bis 64 Bit-Wörter
* Leistung bei 25 MHz Takt > 5 MIPS
* Coprozessorinterface der 680xx-Serie - besonders interessant ist hier der 68882, eine Fließkommaarithmetikeinheit, die parallel zum 68030 arbeitet (1,5-fache Geschwindigkeit des 68881 bei gleicher Taktfrequenz).
* HCMOS-Technologie (Schnell und sparsam)

Literatur:

[1] Datenblatt MC 68030, Motorola GmbH
[2] Datenblatt MC 68882, Motorola GmbH
[3] Thomas L. Johnson, "The RISC/CISC Melting Pot" Byte-Magazin 4/87 S. 153 ff., McGraw-Hill

Kurzdaten des T800:

* Hardware-64-Bit-Fließkommaeinheit nach IEEE 754
* Leistung bei 20 MHz: 1,5 MFlops
* 32-Bit-Architektur mit 10 MIPS Leistung
* Pin- und softwarekompatibel zum T414
* 4 KByte on chip RAM mit 80 MByte/s Datenrate (im on chip RAM)
* 4 GByte direkt adressierbarer Speicher, Transferrate 26,67 MByte/s
* Grafikunterstützung (2D-Blockkopien in Hardware)
* 4 Serielle Links mit 5/10 oder 20 MBaud
* Hardware-Scheduler für Multitasking
* Interner Timer für Echtzeitanwendungen
* DRAM-Controler auf dem Chip

Alle Geschwindigkeitsangaben für die 20 MHz-Version.

Literatur:

[1] IMS T800 Datenblatt, Inmos GmbH
[2] IMS C004 Datenblatt, Inmos GmbH
[3] Peter Eckelmann, "Transputer der 2. Generation" Elektronik Heft 18 S. 61 ff., Heft 19 S. 129 ff., Heft 20 S. 86ff. Franzis Verlag
[4] "The Transputer Instruction Set - A Compiler Writers Guide", Inmos GmbH

UNLIMITED

PREISWERT WOLLEN WIR SEIN, DESHALB KOMMT KEINE TEURE ANZEIGE MEHR REIN!

RESTPOSTEN

Garrison I oder II. dtsch.	39,- DM
Bad Cat Katzenolympiade dtsch.	25,- DM
Spaceport dtsch.	20,- DM
Mission Elevator dtsch.	20,- DM
Powerplay	50,- DM
Pageflipper dtsch.	50,- DM
Comicsetter	120,- DM
Arazoks Tomb	50,- DM
Strike Force Harrier	50,- DM
King of Chicago	60,- DM
Space Baller	15,- DM

RIESENAUSWAHL ZU TIEFSTPREISEN!
FORDERN SIE UNSERE KOSTENLOSE LISTE AN!
TELEFONISCHE BESTELLANNAHME RUND UM DIE UHR

Telefon: 06121 / 543848

UNLIMITED M.Hottenbacher, Kehrstrasse 23, 6200 Wiesbaden Kein Parteienverkehr!

GENLOCK 8802	DM 998,-
PERFECT VISION	
PAL Color Digitizer	DM 498,-
POLAROID PALETTE	
mit IMPRINT	DM 6750,-
Vorführmodelle	DM 4500,-
DIA BELICHTUNGEN:	
1 Stk.	DM 4,95
ab 36 Stk.	DM 3,31
DIN A4 Rank Xerox	
Overheadfolie	DM 15,-
VIDEO EFFECTS 3D	
mit dt. Handbuch	DM 379,-
3 DEMON	
mit dt. Handbuch	DM 215,-
TURBO SILVER	
mit dt. Handbuch	DM 359,-
THE DIRECTOR	
mit dt. Handbuch	DM 159,-
MODELER 3D	
mit dt. Handbuch	DM 218,-
VIDEOSCAPE 3D V 2.0	
mit dt. Handbuch	DM 379,-
PRO VIDEO PLUS	
mit dt. Handbuch	DM 598,-
ANIMATE 3D	
mit dt. Handbuch	DM 259,-
SCULPT 3D	
mit dt. Handbuch	DM 198,-
PHOTON PAINT	
mit dt. Handbuch	DM 249,-
PIXMATE	
mit dt. Handbuch	DM 149,-
STUDIO MAGIC	
mit dt. Handbuch	DM 129,-
SONIX	
mit dt. Handbuch	DM 175,-
FANTAVISION	DM 139,-
AUDIO MASTER	
mit dt. Handbuch	DM 134,-
PERFECT SOUND	
Stereo Digitizer	
mit dt. Handbuch	DM 198,-
DELX. MUSIC C. SET	
mit dt. Handbuch	DM 179,-
AZTEC C V 3.6 Prof.	
mit dt. Handbuch	DM 499,-
AZTEC C V 3.6 DEV.	
mit dt. Handbuch	DM 655,-
dt. Handbuch solo	DM 138,-
ZING KEYS	
mit dt. Handbuch	DM 129,-
DIGA	
mit dt. Handbuch	DM 169,-
CREDITS	
Scrollingprogramm	DM 128,-

Für Besitzer der Originalprogramme gibt es deutsche Handbücher einzeln für DM 39,95

weitere INFO: LOFT POST
video LOFT film HARD & SOFTware GmbH
Fiedlerstr. 22 - 32
BRD 3500 Kassel
tel.: 0561 - 87 33 99 / 87 79 28
fax: 0561 - 87 80 48



MegaTronic-Laufwerke für AMIGA und Atari ST

Preise:		DM
MegaTronic 3.5″	für AMIGA	299,–
MegaTronic 3.5″-digi	für AMIGA	349,–
MegaTronic 5.25″	für AMIGA	379,–
MegaTronic 5.25″-digi	für AMIGA	429,–
MegaTronic 3.5″	für Atari ST	339,–
MegaTronic 3.5″-digi	für Atari ST	379,–
MegaTronic 5.25″	für Atari ST	399,–
MegaTronic 5.25″-digi	für Atari ST	449,–
MegaTronic 3.5″-Doppelstation	für ST	599,–

LOW-COST-Drives für AMIGA und Atari ST – bereits tausendfach bewährt

komplett anschlußfertig, abschaltbar, formschönes Metallgehäuse, 880 KB formatiert unter AMIGA, 720 KB formatiert unter Atari ST, Disk-Change wird erkannt, 6 Monate Garantie. Aufpreis für Busdurchführung: 15,– DM für AMIGA, 25,– DM für Atari

AMIGA	SDN 3.5″ – NEC 1037A	249,–
AMIGA	SDN 3.5″ digital – NEC 1037A	289,–
AMIGA	SDN 5.25″ – TEAC FD 55	299,–
AMIGA	SDN 5.25″ – TEAC FD 55 GFR (HD)	309,–
AMIGA	SDN 5.25″ – NEC 1157C	309,–
AMIGA	SDN 5.25″ digital	339,–

Sonderaktion für AMIGA:
SDN 3.5″ – NEC 1036 A, komplett anschlußfertig, bewährte robuste Mechanik 229,–
SDN 3.5″ intern für A 2000 189,–

Atari	SL 3.5″ – NEC 1037A	265,–
Atari	SL 3.5″ digital	309,–
Atari	SL 5.25″	319,–
Atari	SL 5.25″ digital	369,–

Rohlaufwerke (unmodifiziert):

NEC	1036 A	189,–
NEC	1037 A	195,–
NEC	1157 C	239,–
TEAC	FD 55 FR	229,–
TEAC	FD 55 GFR (HD-Laufwerk)	239,–
Gehäuse	3.5″ AMIGA	19,–
Gehäuse	3.5″ Atari	27,–
Gehäuse	5.25″ AMIGA, Atari	29,–

Festplatten für AMIGA

komplett anschlußfertig, inkl. Controller, Anschlußkabel, Software, Einbau- und Bedienungsanleitung

Harddisk 30 MB für A 2000 intern 799,-
- SEAGATE-Platte, 5.25″
- 65 ms, Datentransfer 150 KB/s

Harddisk 30 MB für A 2000 intern 959,–
- SEAGATE-Platte, 3.5″
- 35 ms, 180 KB/s

Harddisk 30 MB für A 500/1000 949,–
- SEAGATE-Platte, 5.25″
- 65 ms, 150 KB/s
- formschönes Gehäuse

Harddisk 60 MB für A500/1000 1.449,–
- SEAGATE-Platte, 5.25″
- 40 ms, 180 KB/s
- formschönes Gehäuse

Weiter Festplatten mit anderen Kapazitäten auf Anfrage

AMIGA 2000 & Monitor 1084	2350,–
XT-Karte	990,–
AT Karte	auf Anfrage
NEC P2200 dt.	849,–
Star LC10 dt.	625,–
Star LC10 Color dt.	749,–
Epson LQ 500	879,–
Epson LQ 850	1.499,–
Epson LX 800	649,–
Citizen 120 D	489,–
NEC P6 Plus	1.650,–
HP Deskjet Tintenstrahldrucker	2.490,–
Filecard 20 MB	749,–
Seagate ST 225	509,–
Seagate ST 238 R	529,–
Omti 5520	179,–
Omti 5527	199,–

Alle Angebote freibleibend, technische Änderungen vorbehalten.

Fordern Sie unverbindlich unsere Preisliste über Computersysteme, Festplatten, Drucker, Laserdrucker, Monitore, Laufwerke, Netzwerke, optische Speichersysteme... an.

Stalter Computerbedarf · Gartenstr. 17
6670 St. Ingbert · Tel. (06894) 35231

VON WILFRIED SOLBACH

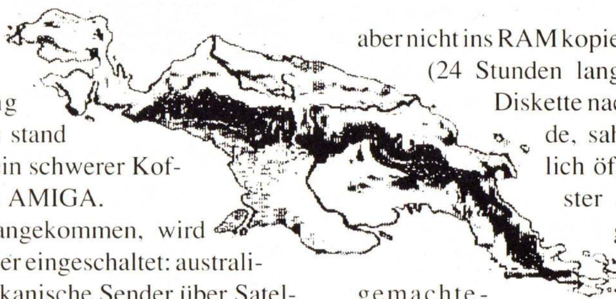
AMIGA IN PAPUA NEUGUINEA



Daltron Electronics in Port Moresby, der Hauptstadt von Papua / Neuguinea

Von uns aus gesehen einer der entlegensten Flecken auf der Landkarte - und ausgerechnet dort soll es ...
... es gibt!

Schon bei der Ankunft auf dem Flughafen der Hauptstadt Port Moresby hatte ich meine erste Begegnung mit einem AMIGA. Ein ziemlich ausgemergelter Drogenhund trabte lustlos über das Transportband, vor dem die Fluggäste der Maschine aus Singapur auf ihre Koffer warteten. Zwischen all den Koffern erschien plötzlich auch ein originalverpackter 500-er, kurz darauf ein 1084 Highscreen-Monitor.



Bei der nächsten Umdrehung des Bandes stand allerdings ein schwerer Koffer auf dem AMIGA.
Im Hotel angekommen, wird der Fernseher eingeschaltet: australische, amerikanische Sender über Satellit (another boring cultural program...), der einheimische Sender EM-TV, ein Videokanal und dann, ich traute meinen Augen nicht, die wohlbekannte blaue AMIGA-workbench mit einem Requester *"Volume copy 2 of Fred has a read/ write error"*.
Des Rätsels Lösung: Dieser 'Infokanal' des Hotels wird von einem AMIGA 500 kontrolliert. Normalerweise läuft auf diesem Kanal eine slide-show (mit TV Text/TV Show erstellt), die den Hotelgästen alllgemeine Informationen über das Hotel vermittelt. Da die slide-show aber nicht ins RAM kopiert war, sondern (24 Stunden lang!) immer von Diskette nachgeladen wurde, sah man schließlich öfter den Requester als die - übrigens sehr professionell gemachte- Informationshow.

Der nächste Morgen führte mich dann zu Daltron Electronics, dem einzigen Commodore (und AMIGA)-Händler in Papua / Neuguinea. Robert Masters, der Technical Sales Manager von Daltron, zeigte mir bereitwillig den ganzen Laden, in dessen Haupt-"showroom" der AMIGA eine prominente Rolle spielt. Besonders hatte es mir eine Hardware-Erweiterung für den 500er angetan: Das Escortsystem der kalifornischen Firma Expansion Technologies wird seitlich

an den AMIGA angesteckt, bietet Platz für eine RAM-Erweiterung, eine Festplatte und ein externes Laufwerk. Oben auf dem soliden, eleganten Metallkasten findet der Monitor seinen Platz.

Das Ganze kostet (unbestückt) etwa 300 Kina - etwa 720 DM. Ein stolzer Preis für einen leeren Kasten, nur sind die Computerpreise in diesem exorbitant teuren Land ebenfalls gesalzen: Daltron verkauft den AMIGA 500 mit 1MB RAM, Farbmonitor sowie einem integrierten Software-Paket für umgerechnet knapp über 3000 DM.

Bob Masters hat auch andere Zahlen parat: von den etwa 160 AMIGA in Papua / Neuguinea (von denen er exakt 98 verkauft hat, der Rest sind Privatimporte über Singapur und Australien) sind etwa 2/3 500er, der Rest 1000er, 2000er gibt es im ganzen Land nur zwei. Einer davon steht bei Bob zu Hause, mit 3MB RAM, einer Festplatte und XT-Karte.

Angesprochen auf die Hauptprobleme sagt er "Schimmel und Viren". Schimmel ist aber kein besonders tückischer Virus, sondern halt ordinärer Schimmel, der sich wegen der hohen Luftfeuchtigkeit auf der magnetisierten Oberfläche der Disketten festsetzt und die Laufwerke ruiniert.

Die meistverkaufte Software in Papua / Neuguinea? Der Commodore-Vertreter will nicht so recht mit der Sprache heraus.

"Wissen Sie, 'Zoom' habe ich schon ein paarmal verkauft."

Schließlich läßt er die Katze aus dem Sack: In Papua / Neuguinea gibt es kein Copyright-Gesetz, jeder kopiert fröhlich und unbehelligt drauflos, der Software-Verkauf geht eher gegen Null.

"Haben Sie schon eine Kopie von 'Acquisition'? Wollen Sie eine?"

Ich wehre verlegen ab.

Zum Thema Viren: SCA und Byte feiern auch in Port Moresby fröhliche Urstände, und des öfteren muß der Mann von Commodore ausrücken, um dem Byte-Bandit auf einer Raubkopie den Garaus zu machen.

"Ich mache so etwas zwar nicht gerne, aber hier kennt jeder AMIGA-Besitzer jeden, und wie schnell hat man da einen schlechten Ruf!"

ENDE



Der Showroom von Daltron Electronics. Das Plakat zeugt von dem immer noch aktuellen C64. Der AMIGA wurde 98 mal verkauft.



Auf der Suche nach einem entlegenen AMIGA. Oft muß auch hierhin der Virenjäger ausrücken, um den Byte Bandit zu vertreiben.

NEUGUINEA



Insel zwischen 0-10° südl. Breite, N/E von Australien

Klima:	Niederschlag im Jahr ca. 2250 mm (zum Vergleich: Frankfurt/M. ca. 600 mm), beständig regenreich, Jahresdurchschnittstemperatur 27,5°C (Frankfurt/M. 9,6°C/Jahr, im Sommer 17.5°C)
Nutzung:	Kokospalmen (Kopra, Öl), Tee, Kaffee, Sisal, Hanf, Kakao
Vegetation:	Regenwald bis 900 m, darüber Nebelwälder, noch höher Grasland
Geomorphologie:	Maokegebirge durchzieht die Insel quer (max. 5029 m) Schneegrenze 4300 m.
Bodenschätze:	Kupfer, Gold, Erdöl
Fläche:	Gesamt 785000 km² (ca. Skandinavien). Das Innere ist unerforscht.
Staaten:	2 Staaten, westlicher Teil (Irian-Jaya) gehört zu Indonesien, östlicher Teil selbständig - Papua / Neuguinea, Hauptstadt Port Moresby
Einwohner:	3,2 Mio., davon 40000 Europäer
Währung:	Kina 1K= ca. 2,40 DM

VON MARTIN SILGERNAGL

KICKUP

ACTION und REACTION oder A wonderfull Ballerspiel



Teil 5

Wiedermal geben wir uns die Ehre: Ich in der Hoffnung, berühmt zu werden - Sie in der Hoffnung, daß der Kurs endlich mal was Sinnvolles bringt. Und wahrscheinlich werden wir wieder beide enttäuscht werden. Trotz dieser Tatsache und trotzdem ich heute beim Friseur war - es muß immer weitergehen!

Was gibt's denn diesmal Neues?", könnten Sie fragen. Und ich könnte sagen, daß ich den "IFF-loader mit Musik" schon halb zum Spiel gemacht hab', obwohl mich widrigste Umstände (Grippe, Physiklehrer und sogar Muskelkater) davon abzuhalten versuchten. Ich könnte Ihnen weiterhin erzählen, daß man schon ein Raumschiff bewegen kann, daß man schießen kann, daß im Hintergrund hunderttausende von Sternen durch die Gegend rasen, daß der IFF-loader nun doch seinen Speeder bekommen hat (3 Sekunden für mein Titelbild braucht er, ehrlich) und überhaupt, daß ALLES ganz, ganz toll geworden ist. Aber Sie wollen das ganz bestimmt nicht wissen.

Was Sie aber sicher gerne hören: Ich habe einen Fehler gemacht! Ja, lachen Sie nur, denn Schadenfreude ist der schönste Regenschirm. Ich habe bei der Musikroutine nicht bedacht, daß die aktuelle Periodentabelle auch viermal abgelegt werden muß, wenn man verschiedene Wellenformen gleichzeitig benutzt. Deshalb drei Änderungen in der Musikroutine: In den Abschnitten CHK_NEW_PERIOD und CHK_NEW_WAVE wurde die Adressierungsart angeglichen, und bei den Variablen wurde halt noch was geändert... Aber ich habe den Fehler gestanden und fordere härteste Bestrafung zu meiner Läuterung.

Konzept macht sich immer gut!

Aber vielleicht sollten wir keinen anomalen Such-/Besprech-Algorihthmus verwenden (eben ist mir UBM-Text abgestürzt und ich hab' eine halbe Stunde gebraucht, um den Text mit dem Debugger zu retten. Ich hatte ihn bis jetzt noch nicht abgespeichert.) und deshalb erst mal wieder das Konzept ausbreiten: wird während des Titelbildes der Joystickknopf gedrückt, fängt das eigentliche Spiel an, und das Programm läßt den AMIGA ständig zwischen zwei Bildschirmen hin- und herschalten. "Spinnt der jetzt ganz und gar? Warum denn das?" Simpel: Im einen Bildschirm werden das Raumschiff, die Sterne und die Schüsse aufgebaut und die alten Objekte gelöscht, der andere Bildschirm wird dargestellt. So verhindert man das Flackern der Objekte, wenn Sie dargestellt werden sollen, aber noch gar nicht ganz fertig aufgebaut sind. Ganz nebenbei: dieses Prinzip von den zwei Bildschirmen nennt man double-buffering, erfunden vor 20 Jahren.

Vielleicht wird Ihnen die Sache klarer, wenn ich (ich spiele den Grünfried) das Programm zusammen mit Drösel (gespielt von mir) von vorne bis hinten durcharbeite:

Jetzt wird's schlimm

Grünfried: Nach ein paar Registern, auf die ich genauer eingehen möchte, wenn ich sie verwende, und nach den bekannten Konstanten...

Drösel: *Eine Konstante ist eine feste Zahl: wie z.B. das Verhältnis zwischen Dummheit des Landwirts und der Größe der Kartoffeln.*

Grünfried: (mit Nachdruck) ...kommen Konstanten, die in unserem Schema total neu sind.

Drösel: *Warum die nun wieder?*

Grünfried: Ganz einfach: weil ich nicht davon ausgehe (weil ich Humanist bin), daß Dein Raumschiff genauso groß sein wird, wie meins, und daß Du Dein Spielfeld genauso groß machst wie ich, kannst Du über diese Konstanten leicht bestimmte Größen ändern, ohne daß Du das Programm nach Stellen durchforsten mußt, an denen diese Größe verwendet wird. Welche Größe die Konstante darstellt, ist am Kommentar hinter der Zeile leicht abzulesen. Leichter geht's nimmer. Und wenn doch was unklar ist: Einfach gucken, wo das Ding wann und wie verwendet wird.

Drösel: *Ach so, na dann.*

Grünfried: Etwas weiter (um das Label COPY_BOBS herum) steht dann eine simple Kopierroutine, die die Bobs ins CHIP-RAM legt, damit sie der Blitter ansprechen kann (zum Blitter komm' ich ja noch... immer mit der Panik).

Drösel: *Ach, arbeitet der vieleicht auch per DMA-Kanal?*

Grünfried: (wiederum mit Nachdruck) Dazu komm' ich noch. (flüsternd zu Drösel) Willst Du uns denn lächerlich machen, vor all den Leuten?

Grünfried: (mit normaler Stimme) Was ich persönlich fürchterlich finde, ist, daß mir langsam aber sicher die Satzanfänge wie "Etwas weiter", "Dann", "Als nächstes" usw. ausgehen. Damit die Sache nicht ins Peinliche abrutscht, übernehme ich ab jetzt keine Garantie für originelle Einfälle. Ätsch.

```
; ----------------------------------------------------------------
;
;         the official Kickstart-game, part four!
;
;         code and comments by Martin Silbernagl
;
;         graphics by Sebastian Dosch
;
;         last revision: 16/12/1988
;
; ----------------------------------------------------------------

DMACONR         EQU     $DFF002
JOY0DAT         EQU     $DFF00A
JOY1DAT         EQU     $DFF00C
INTREQR         EQU     $DFF01E
INTENAR         EQU     $DFF01C
BLTCON0         EQU     $DFF040
BLTCON1         EQU     $DFF042
BLTAFWM         EQU     $DFF044
BLTALWM         EQU     $DFF046
BLTCPTH         EQU     $DFF048
BLTBPTH         EQU     $DFF04C
BLTAPTH         EQU     $DFF050
BLTDPTH         EQU     $DFF054
BLTSIZE         EQU     $DFF058
BLTCMOD         EQU     $DFF060
BLTBMOD         EQU     $DFF062
BLTAMOD         EQU     $DFF064
BLTDMOD         EQU     $DFF066
DIWSTRT         EQU     $DFF08E
DIWSTOP         EQU     $DFF090

[...]

BPL2MOD         EQU     $DFF10A
COLOR00         EQU     $DFF180
VHPOSR          EQU     $DFF006

CIAA_PRA EQU    $BFE001
CIAA_TALO       EQU     $BFE401

[...]

_LVOSeek        EQU     -$042
_LVODisplayAlert EQU    -$05a   ; intuition LVOs

TITLE_P  EQU    40*200             ; Größe einer Titelbild-Plane
TITLE    EQU    TITLE_P*5          ; Größe des Titelbildes
FIELD_B  EQU    44                 ; Bytes pro Zeile beim Titelbild
FIELD_H  EQU    288                ; Höhe des Spielfeldes
FIELD_P  EQU    FIELD_B*FIELD_H    ; Größe einer Spielfeld-Plane
FIELD_O  EQU    FIELD_P*7          ; Größe eines Spielfeldes
FIELD    EQU    FIELD_O*2          ; Größe der zwei Spielfelder
LINES    EQU    4                  ; Zeilen zwischen zwei Sternen
STARS    EQU    288/LINES          ; maximale Anzahl der Sterne
SHIP_W   EQU    3                  ; Breite des Schiffs in Words
SHIP_B   EQU    SHIP_W*2           ; Breite des Schiffs in Bytes
SHIP_H   EQU    18                 ; Höhe des Schiffs
SHIP_P   EQU    SHIP_H*SHIP_B      ; Größe einer Plane
SH_BLIT  EQU    SHIP_H*64+SHIP_W   ; Blitter"SIZE" des Schiffs
XSPEED   EQU    4                  ; Geschwindigkeit des Raumschiffs
YSPEED   EQU    3                  ; Geschwindigkeit des Raumschiffs
LENGTH   EQU    8192               ; Buffergröße festlegen
SHOOT_M  EQU    24                 ; Maximale Anzahl der Schüsse
SHOOT_W  EQU    2                  ; Breite des Schusses in Words
SHOOT_B  EQU    SHOOT_W*2          ; Breite des Schusses in Bytes
SHOOT_H  EQU    1                  ; Höhe des Schusses
SHOOT_P  EQU    SHOOT_H*SHOOT_B    ; Größe einer Plane
SHOOT_X  EQU    (SHIP_W-1)*16      ; rel. Pos. des Schusses: X-Wert
SHOOT_Y  EQU    SHIP_H/2           ; rel. Pos. des Schusses: Y-Wert
SHOOT_S  EQU    12                 ; Geschwindigkeit der Schüsse
SO_BLIT  EQU    SHOOT_H*64+SHOOT_W  ; Blitter"SIZE" des Schusses
BOBS     EQU    6*SHIP_P+6*SHOOT_P  ; Speicher bedarf der
                                      Blitterobjs.
MAXX_SO  EQU    8*(FIELD_B-SHOOT_B+2)-1 ; Maximaler X-Bereich des
                                          Schusses
MAXX     EQU    8*(FIELD_B-SHIP_B+2)-1  ; Maximale X-Position des
                                          Schusses
MAXY     EQU    (FIELD_H-SHIP_H)-1      ; Maximale Y-Position des
                                          Schusses

START:
```

Drösel: *Toll!*

Grünfried: Daraufhin (schön, wa?) folgt ein Einsprung in die Routine mit dem seltsamen Namen INIT_STARS. Wie angesprochen, flitzt im Background des Spiels eine halbe Milchstraße an Sternen durch die Gegend. Diese Sterne tun das von rechts nach links, um den trügerischen Eindruck beim Zuschauer zu erwecken, er fliege von links nach rechts durch den Weltenraum. Um diesen Eindruck zu verstärken, sollen die Sterne mit verschiedenen Geschwindigkeiten durch die Einöde des Weltenraums fetzen. Denn langsame scheinen hinten zu liegen und schnelle vorne.

Drösel: *(hastig und leise) Ich persönlich finde, daß die langsamen hinten liegen und schnell sind und die Schnellen auch hinten liegen, aber schneller sind.*

Grünfried: (D einen bösen Blick zuwerfend) Mit INIT_STARS hat das Ganze soviel zu tun, daß (Anm. des Autors: eben ist UBM-Text schon wieder abgestürzt) die vielen Sterne von zwei Routinen (MOVE_STARS und DEL_STARS) verwaltet werden, die eine Menge Variablen haben, die wiederum von INIT_STARS gesetzt werden müssen, ehe diese Routinen das erste Mal angesprungen werden.

Drösel: *Klar, ey, du, ey, echt, ey.*

Grünfried: Schau'n wir uns die Sternbewegungs- und Löschroutinen mal genauer an (die stehen irgendwo hinter den Musikdaten), wenn wir schon mal dabei sind. DEL_STARS löscht die Sterne, die MOVE_STARS auf den Bildschirm malt. Wichtig zu wissen dabei:

- Pro Zeile gibt es nur einen Stern.
- Ein Stern ist eigentlich nur ein Bit, das in der ersten Plane gesetzt wird.
- Die Konstante STARS gibt die Anzahl der Sterne an, die auf dem Bildschirm zu sehen sind.
- Ab dem Label STARS_ADR stehen die Adressen an, die die Sterne beim letzten Mal gesetzt worden sind.
- Ab dem Label STARS_DEL stehen die Adressen, an die die Sterne vor

```
 MOVEM.L D0-D7/A0-A6,-(A7)        ; Alle Register auf den Stack legen
 MOVEA.L 4,A6                     ; Adresse der exec-library holen
 MOVE.L  #TITLE+FIELD+BOBS,D0     ; Speichergröße nach D0
 MOVE.L  #ADSR1,D1                ; Ende der letzten Wellenform
 SUB.L   #WAVE1,D1                ; davon Anfang der ersten abziehen
[...]
 MOVE.L  D0,FIELD_BASE            ; Adresse des Speichers abspeichern
 ADDI.L  #FIELD,D0                ; Größe des Bildes dazurechnen
 MOVE.L  D0,BOB_BASE              ; Adresse des Speichers abspeichern
 ADDI.L  #BOBS,D0                 ; Größe der Blitter-Objects dazu
 MOVE.L  D0,WAVE_BASE             ; Adresse des Speichers abspeichern
 MOVE.W  #BOBS,D0                 ; Speicherbedarf der BOBS
 LEA     SHIP_DATA,A0             ; Anfang der Daten
 MOVEA.L BOB_BASE,A1              ; Adresse des Chip-Mems
COPY_BOBS:
 MOVE.B  (A0)+,(A1)+              ; ein Byte kopieren
 DBRA    D0,COPY_BOBS             ; Schleife machen
 MOVE.L  #ADSR1,D0                ; Ende der letzten Wellenform
 SUB.L   #WAVE1,D0                ; davon Anfang der ersten abziehen
[...]
LOOP1:   MOVE.L  (A1)+,(A0)+      ; jeweils einen Vektor kopieren
 DBRA    D0,LOOP1                 ; D0=-1? Wenn nicht, dann loop1
 JSR     INIT_STARS               ; Tabellen aufbauen
 JSR     WAIT_BLT                 ; auf möglichen Blit vom BS warten
MAIN:
 MOVE.L  #MAIN,$0020              ; MAIN to Priv._Viol._Vektor
 MOVE.W  #$2700,SR                ; Supervisor=1, Interruptlevel=7
[...]
 MOVE.B  #$8b,CIAA_ICR            ; SP, TA und TB Interrupt erlauben
 MOVE.W  #$00ff,ADKCON            ; keine Modulationen
 MOVE.W  #$ffff,BLTAFWM           ; Blitter: keine Maske beim 1. Word
 MOVE.W  #$ffff,BLTALWM           ; Blitter: keine Maske beim letzte W.
 MOVE.L  FIELD_BASE,A0            ; Adresse des Spielfeldes holen
 MOVE.L  A0,FIELD_BASE1           ; Adresse des ersten Spielfeldes
 MOVE.L  A0,SH_LAST               ; Dummy-Adresse für den Blitter
 ADDA.L  #FIELD_O,A0              ; Platz für ein Feld dazu
 MOVE.L  A0,FIELD_BASE2           ; Adresse des zweiten Spielfeldes
 MOVE.L  A0,SH_BE_LAST            ; Dummy-Adresse für den Blitter
 MOVE.W  #$2000,SR                ; Supervisor=1, Interruptlevel=0
START_OVER:
[...]
 MOVE.W  #$7ff0,DMACON            ; Alle DMA ausschalten
 MOVE.W  #$8300,DMACON            ; DMAEN und BPLEN anschalten
 LEA     TITLE_CMAP,A0            ; Adresse der Quellcolormap
 JSR     COPY_COLORS              ; Farben setzen
 MOVE.W  #$5200,BPLCON0           ; 5 Bitplanes und Colorburst
 MOVE.W  #$0000,BPLCON1           ; Scrollregister auf Null setzen
[...]
 MOVE.W  #$0038,DDFSTRT           ; data-fetch-start definieren
 MOVE.W  #$00d0,DDFSTOP           ; data-fetch-start definieren
 JSR     WAIT_JOYSTICK            ; auf Knopfdruck warten
MAIN_INIT:
 MOVE.L  #MAIN_LOOP,MAIN_JMP+2    ; Vektor der Hauptschleife zurück
                                  ; setzen
 MOVE.L  #PORTS_M,$68             ; Vektor Int.-Level 2 auf PORTS_M
 MOVE.L  #VERTB_M,$6C             ; Vektor Int.-Level 3 auf VERTB_M
 MOVE.W  #$7fff,INTENA            ; Alle Ints. ausschalten
 MOVE.W  #$C028,INTENA            ; PORTS- und VERB-Int. erlauben
 MOVE.W  #$7ff0,DMACON            ; Alle DMA ausschalten
```

zwei Videobildern gesetzt worden sind. Hier sind also die Adressen, die diesmal wieder gelöscht werden sollen. Klar sollte sogar Dir jetzt sein, wie die DEL_STARS-Routine funktioniert.

Drösel: *Äh... Ja...*

Grünfried: (Drösel nicht zu Worte kommen lassend) Eine Schleife mit STARS Durchgängen, die sich die nächste Adresse aus STARS_DEL holt, dort eine Null reinschreibt und dann anstelle dieser Adresse die entsprechende Adresse von STARS_ADR dort hineinschreibt, damit dort beim nächsten Mal dasselbe passiert. Alles verstanden?

Drösel: *Nein.*

Grünfried: Na ja, macht ja nix.

Grünfried: (Wenige Jahrhunderte später) Genau umgekehrt funktioniert jetzt die Routine MOVE_STARS. Nur braucht die noch mehr Variablen: STARS_POS enthält die x-Position des Sterns (bei mir zwischen 0 und 351), und STARS_SPEED enthält einen Wert, der bei jedem Durchlauf (also jedem Videobild) von der x-Position abgezogen wird.

Drösel: *Also (logisch) ist dieser Wert die Geschwindigkeit.*

Grünfried: Ja, Bravo... und was macht MOVE_STARS genau?

Drösel: *Grundadresse des Bildschirms holen - Schleife mit STARS Durchläufen, in der über die Position die Adresse des Sterns berechnet wird, in die er dann auch geschrieben wird.*

Grünfried: Was aber, wenn ein Stern den linken Rand erreicht hat?

Drösel: *Noooch einfacher: Die x-Position dieses Sterns wird wieder auf 351 (linken Rand) gestellt, und eine neue, zufällig gewählte Geschwindigkeit wird in die entsprechende Stelle von STARS_SPEED geschrieben...*

Grünfried: (mit Verzückung, den Mund spitzend) Eine hervorzügliche

```
 MOVE.W  #$8740,DMACON              ; BLTPRI,DMAEN,BPLEN & BLTEN setzen
 LEA     MAIN_CMAP,A0               ; Adresse der Quellcolormap
 JSR     COPY_COLORS                ; Farben setzen
 MOVE.W  #$5200,BPLCON0             ; 5 Bitplanes und Colorburst
 MOVE.W  #$0000,BPLCON1             ; Scrollregister auf Null setzen
 MOVE.W  #$0000,BPLCON2             ; Prioritäten normal setzen
 MOVE.W  #0,BPL1MOD                 ; kein Modulo setzen
 MOVE.W  #0,BPL2MOD                 ; kein Modulo setzen
 MOVE.W  #$1671,DIWSTRT             ; linke, obere Ecke setzen
 MOVE.W  #$36d1,DIWSTOP             ; rechte, untere Ecke setzen
 MOVE.W  #$0030,DDFSTRT             ; data-fetch-start definieren
 MOVE.W  #$00d8,DDFSTOP             ; data-fetch-start definieren
MAIN_LOOP:
 MOVEA.L FIELD_BASE1,A0                 ; Adresse des ersten Spielfeldes
 MOVE.L  FIELD_BASE2,FIELD_BASE1        ; Adresse des zweiten kopieren
 MOVE.L  A0,FIELD_BASE2                 ; Vertauschung vollenden
 JSR     DEL_STARS                      ; Sterne  löschen
 JSR     DEL_SHOOTS                     ; Schüsse löschen
 JSR     DEL_SHIP                       ; Schiff  löschen
 JSR     MOVE_STARS                     ; Sterne  bewegen
 JSR     MOVE_SHOOTS                    ; Schüsse bewegen
 JSR     MOVE_SHIP                      ; Schiff  bewegen
 JSR     TEST_BUTTONS                   ; Maus- & Joystickknöpfe prüfen
 MOVE.L  #MAIN_WAIT,MAIN_WAIT+2         ; Vektor setzen
MAIN_WAIT:
 JMP     MAIN_WAIT                      ; auf VERTB warten
MAIN_JMP:
 JMP     MAIN_LOOP                      ; Hauptschleife
[...]
; ------------------------------------------------------------------
ENDING:
 JSR     WAIT_BLT                   ; auf Ende eines mögl. Blits warten
 MOVE.W  #$2700,SR                  ; auf höchste Int. Ebene setzen
 ORI.W   #$8000,INTENA_BUF          ; Set-Bit im Zwischenspeicher setzen
[...]
 MOVEA.L 4,A6                       ; Basis der exec-Library holen
 MOVEA.L TITLE_BASE,A1              ; Basis nach A1 holen
 MOVE.L  #TITLE+FIELD+BOBS,D0       ; Grösse nach D0
 MOVE.L  #ADSR1,D1                  ; Ende der letzten Wellenform
 SUB.L   #WAVE1,D1                  ; davon Anfang der ersten abziehen
[...]
 MOVE.W  (A7)+,D0                   ; D0 vom Stack
 RTE                                ; Rückkehr auf andere Ebene
PORTS_M:                            ; Server von Interrupt-Level 2
 MOVE.W  #$0008,INTREQ              ; Interrupt-Bit wieder zurücksetzen
 MOVE.W  D0,-(A7)                   ; D0 auf den Stack
 MOVE.B  CIAA_ICR,D0                ; Interruptregister auslesen
 BTST    #3,D0                      ; SP-Bit gesetzt?
 BEQ.S   NO_SP_M                    ; wenn nicht, dann andere Bits
 JSR     KEY_WORX                   ; Tastatur abfragen
NO_SP_M:
 MOVE.W  (A7)+,D0                   ; D0 vom Stack
 RTE                                ; Rückkehr auf andere Ebene
KEY_WORX:
 MOVEM.L D0-D2/A0,-(A7)             ; D0-D2 und A0 auf den Stack
[...]
 MOVEM.L (A7)+,D0-D1/A0             ; Register vom Stack holen
 RTE
VERTB_M:
 MOVE.W  #$0020,INTREQ              ; Request-Bit loeschen
```

```
  MOVEM.L D0-D3/A0,-(A7)      ; Register auf den Stack legen
  MOVEQ   #4,D0                ; 5 Bitmapointer sind zu setzen
  MOVE.L  FIELD_BASE1,D1       ; Adresse des Titelbildes
  LEA     BPL1PT,A0            ; Adresse des ersten Registers
SET_FIELD:
  MOVE.L  D1,(A0)+             ; Adresse der Plane setzen
  ADDI.L  #FIELD_P,D1          ; eine Plane dazuaddieren
  DBRA    D0,SET_FIELD         ; naechste plane setzen
  MOVE.L  #MAIN_JMP,MAIN_WAIT+2 ; Vektor setzen

  MOVEM.L (A7)+,D0-D3/A0       ; Register vom Stack holen
  RTE

[...]

  MOVE.L  D0,D3                ; Anzahl der zu lesenden Bytes
  MOVE.L  D0,D4                ; Kopie der Anzahl machen
  JSR     BUFFER_READ          ; und lesen
  CMP.L   D4,D0                ; gewuenschte Anzahl gelesen?
  BEQ.S   READ_DATA_OK         ; wenn ja, dann weiter

[...]

  MOVE.L  D0,D2              ; Anz. der zu ueberlesenden Bytes
  MOVEQ   #0,D3              ; Von der aktuellen r/w-position aus
  JSR     BUFFER_SEEK        ; Bytes ueberlesen
  CMPI.L  #-1,D0             ; gewuenschte Anzahl ueberlesen?
  BNE.S   SKIP_DATA_OK       ; wenn ja, dann weiter

[...]

OPEN_IFF_OK:
  MOVE.L  D0,FILE              ; file-handle abspeichern
  JSR     INIT_BUFFER          ; Buffer allocieren
  MOVEM.L (A7)+,D0-D2/A0-A1/A6 ; Register vom Stack retten
  RTS                          ; Rueckkehr
CLOSE_IFF:
  JSR     FREE_BUFFER        ; Speicher freigeben
  MOVE.L  DOS_BASE,A6        ; pointer auf die DOSch-library holen
  MOVE.L  FILE,D1            ; file-handle nach D1 laden

[...]

  JSR     ERRORS             ; und ausgeben
  JMP     CLOSE_IFF          ; IFF-file schliessen

BUFFER_READ:                 ; D1:file D2:Adr. D3:Länge
  MOVEM.L D1-D5/A2-A3/A6,-(A7) ; Register retten
  MOVE.L  D2,D4              ; Zwischenspeicher anlegen
  MOVE.L  D3,D5              ; Zwischenspeicher anlegen
  MOVE.L  REST,D0            ; Anzahl der Bytes im Buffer
  CMP.L   D3,D0              ; noch genügend Daten vorhanden?
  BGE.S   BUFFER_COPY_NOW    ; dann nächsten Teil überspringen
  MOVEA.L POSITION,A2        ; aktuelle Bufferposition holen
  MOVEA.L BUFFER,A3          ; Adresse des Buffers holen
  TST.L   D0                 ; REST genau 0?
  BEQ.S   BREAD_NO_COPY      ; dann ist der Buffer ganz leer
  SUB.L   #1,D0              ; D0 zum Schleifenzähler machen
BREAD_COPY:
  MOVE.B  (A2)+,(A3)+        ; ein Byte kopieren
  DBRA    D0,BREAD_COPY      ; Restbuffer kopieren
BREAD_NO_COPY:
  MOVE.L  A3,D2              ; Adresse des leeren (Rest-) Buffers
  MOVE.L  #LENGTH,D3         ; Länge des Buffers holen
  SUB.L   REST,D3            ; davon Länge des vollen abziehen
  MOVEA.L DOS_BASE,A6        ; Adresse der DOS-library holen
  JSR     _LVORead(A6)       ; Buffer vollmachen
  MOVE.L  #LENGTH,REST       ; der Buffer ist wieder voll
  MOVE.L  BUFFER,POSITION    ; auf Anfangsposition setzen
BUFFER_COPY_NOW:
  MOVEA.L POSITION,A2        ; aktuelle Bufferposition holen
  MOVE.L  D5,D0              ; D5 kopieren
  SUB.L   #1,D5              ; D5 (Länge) zum Schleifenzähler
  MOVE.L  D4,A3              ; gewünschte Adresse nach A1
BUFFER_COPY:
  MOVE.B  (A2)+,(A3)+        ; ein Datenbyte kopieren
  DBRA    D5,BUFFER_COPY     ; Schleife machen
  MOVE.L  A2,POSITION        ; neue Adresse abspeichern
  SUB.L   D0,REST            ; Anzahl der gelesenen Bytes abziehen
  MOVEM.L (A7)+,D1-D5/A2-A3/A6 ; Register zurück
  RTS
```

Antwort. Aber woher soll ein Computer eine zufällige Zahl nehmen? Was für einen Menschen eine der leichtesten Aufgaben ist, scheint auf einem Computer fast unlösbar zu sein.

Drösel: *(verlegen) Also, ma kann ja net alles wissä, odä?*

Grünfried: (enttäuscht) Nun, in den verschiedensten Computerzeitschriften (in der KICKSTART noch nicht - bis genau jetzt) sind schon die verschiedensten Arten erörtert worden, eine zufällige Zahl zu erzeugen. Einige Beispiele:

- Hardwarezufallsgeneratoren (zu aufwendig)

- Berechnung einer Zufallszahl mit Hilfe einer mathematischen Formel (zu langsam)

- Berechnung mit Hilfe des aktuellen Stackinhalts (zu unzufällig)

- Berechnung durch die horizontale und vertikale Position des Videostrahls (ist o.k.)

Das letzte Beispiel müßtest Du eigentlich selbst umsetzen können, denn das Register VHPOSR, das die entsprechenden Werte liefert, hatten wir schon in der ersten Folge (uoh, 5 Monate is' das schon wieder her). Wenn Du's nicht schaffst, macht das aber auch so gut wie gar nix, denn ab dem Label FAST_RAND ist die Funktion zu finden. Allerdings hat sie auch ziemliche Nachteile: Sie liefert nur Werte von 0 bis 255. Aber wer kann schon so weit zählen?

Drösel: *Na ich!*

Grünfried: Das war ein Witz!

Drösel: *(beschämt) Ach so.*

Grünfried: So. Und wenn Dir das alles zu hoch ist, dann laß die blöden Sterne halt weg - kosten eh genug CPU-Rechenzeit. Was viel wichtiger ist: der BLITTER (täterätä! Konfetti)

Drösel: *Alles schön und gut, aber wer ist ein Blitter?*

```
BUFFER_SEEK:                          ; D1:file D2:Länge
  MOVEM.L D1-D4/A2-A3/A6,-(A7)          ; Register retten
  MOVE.L  D2,D4                         ; Zwischenspeicher anlegen
  MOVE.L  REST,D0                       ; Anzahl der Bytes im Buffer
  CMP.L   D2,D0                         ; noch genügend Daten vorhanden?
  BGE.S   SBUFFER_COPY_NOW              ; dann nächsten Teil überspringen
  MOVEA.L POSITION,A2                   ; aktuelle Bufferposition holen
  MOVEA.L BUFFER,A3                     ; Adresse des Buffers holen
  TST.L   D0                            ; REST genau 0?
  BEQ.S   BSEEK_NO_COPY                 ; dann ist der Buffer ganz leer
  SUB.L   #1,D0                         ; D0 zum Schleifenzähler machen
BSEEK_COPY:
  MOVE.B  (A2)+,(A3)+                 ; ein Byte kopieren
  DBRA    D0,BSEEK_COPY               ; Restbuffer kopieren
BSEEK_NO_COPY:
  MOVE.L  A3,D2                       ; Adresse des leeren (Rest-) Buffers
  MOVE.L  #LENGTH,D3                  ; Länge des Buffers holen
  SUB.L   REST,D3                     ; davon Länge des vollen abziehen
  MOVEA.L DOS_BASE,A6                 ; Adresse der DOS-library holen
  JSR     _LVORead(A6)                ; Buffer vollmachen
  MOVE.L  #LENGTH,REST                ; der Buffer ist wieder voll
  MOVE.L  BUFFER,POSITION             ; auf Anfangsposition setzen
SBUFFER_COPY_NOW:
  MOVE.L  D4,D0                       ; D5 kopieren
  ADD.L   D0,POSITION                 ; neue Adresse berechnen
  SUB.L   D0,REST                     ; Anzahl der gelesenen Bytes abziehen
  MOVEM.L (A7)+,D1-D4/A2-A3/A6        ; Register zurück
  RTS

INIT_BUFFER:
  MOVEM.L D0-D1/A0-A1/A6,-(A7)          ; Register retten
  MOVE.L  #LENGTH,D0                    ; Buffergröße festsetzen
  MOVEQ   #0,D1                         ; keinen besonderen Speicher
  MOVE.L  4,A6                          ; Adresse der exec-library
  JSR     _LVOAllocMem(A6)              ; Buffer holen
  TST.L   D0                            ; Speicher bekommen?
  BNE.S   INIT_BUFFER_OK                ; wenn ja, dann weiter
  MOVEM.L (A7)+,D0-D1/A0-A1/A6          ; Register zurückholen
  MOVEQ   #8,D0                         ; Fehler Nummer 8
  JSR     IFF_ERROR                     ; Fehler ausgeben
INIT_BUFFER_OK:
  MOVE.L  D0,BUFFER                     ; Adresse des Buffers abspeichern
  MOVE.L  D0,POSITION                   ; Anfangsposition setzen
  CLR.L   REST                          ; keine Zeichen mehr im Puffer
  MOVEM.L (A7)+,D0-D1/A0-A1/A6          ; Register zurückholen
  RTS

FREE_BUFFER:
  MOVEM.L D0-D1/A0-A1/A6,-(A7)          ; kennst DU Thomas Breunig?
  MOVE.L  BUFFER,A1                     ; Adresse des Buffers holen
  BEQ.S   NO_FREE                       ; wenn nicht, dann zurück
  MOVE.L  #LENGTH,D0                    ; Größe nach D0
  MOVE.L  4,A6                          ; Adresse der exec-library holen
  JSR     _LVOFreeMem(a6)               ; Speicher freigeben
NO_FREE:
  MOVEM.L (A7)+,D0-D1/A0-A1/A6          ; hier steht nichts.
  RTS

BMHD:
  DC.W    0,0                           ; Breite und Hoehe

[...]

IFF_COLORS:
  DS.W    32                            ; Platz fuer 32 RGB-Werte


; ----------------------------------------------------------------------
;
; Ab hier stehen verschiedene kleine Routinen
;
; ----------------------------------------------------------------------

COPY_COLORS:                            ; Farbregister setzen
  MOVEM.L D0/A0-A1,-(A7)                ; Register retten
  LEA     COLOR00,A1                    ; Basisadresse der Farbregister
  MOVEQ   #31,D0                        ; 32 mal kopieren
COPY_CMAP2:
  MOVE.W  (A0)+,(A1)+                   ; eine Farbe kopieren
  DBRA    D0,COPY_CMAP2                 ; Schleife machen
  MOVEM.L (A7)+,D0/A0-A1                ; Register retten
  RTS
```

Grünfried: Der Blitter ist eine außerordentlich nützliche und leistungsfähige Einrichtung, die aber leider knapp zehnmal zu langsam ist.

Drösel: *(mit ironischem Unterton) Toll. Jetzt weiß ich mehr.*

Grünfried: Ich war ja noch nicht fertig! Der Blitter dient im AMIGA dazu, Daten zu bewegen. Also alles fast ohne die CPU und sogar während die was anderes macht (Im Regelfall berechnet die CPU dann gerade die nächste Adresse, zu der die Daten bewegt werden sollen). Wie Speicherzugriff ohne die CPU funktioniert, mußt Du aber wissen, Drösel, das hatten wir schon bei der Bitmap und beim Sound.

Drösel: *DNS..äh DNA..äh DMA!*

Grünfried: Jawoll! Also müssen wir dem Blitter die DMA erlauben, indem wir das entsprechende Bit im DMACON setzen [Anm. des Autors: seltsame Begebenheit -beim 500er (ob bei allen, weiß ich nicht) ist die Blitter-DMA immer an!]. In unserem Spiel...

Martin: *Also eigentlich ist's ja meins.*

Grünfried: ...geschieht dies nach dem Label MAIN_INIT, ab dem die Register fürs eigentliche Spiel gesetzt werden. Doch hier wird noch ein weiteres Bit gesetzt mit dem Namen BLTPRI. Ist das auf Eins gesetzt, hat der Blitter vor der CPU Priorität und umgekehrt umgekehrt.

Drösel: *Worin Priorität? Bei der Besoldung?*

Grünfried: Ernst bitte! Bei der Okkupation des Busses natürlich. Denn wir haben ja nur einen, haha. Und da bei uns der Blitter eh mehr zu tun hat, bekommt er natürlich, was ihm gebührt: Priorität.

Drösel: *Und wie starte ich den Blitter nun?*

Grünfried: Der Reihe nach, bitte. Zuerst müssen wir ja Ziel und Quelle angeben. Und schon kommt wieder eine Besonderheit: Wir haben drei Quellen, die logisch verknüpft zum Ziel kommen.

Welche der vier Kanäle benutzt werden, steht im BLTCON0 (Tab. 1).

Drösel: *Ja, ist es denn überhaupt sinnvoll, keine Quelle oder kein Ziel zu verwenden?*

Grünfried: Aber ja. Aber dazu muß man wieder etwas wissen: Es gibt im DMACONR-Register ein Bit (das dreizehnte), das angibt, ob während eines Blits eine 1 aufgetaucht ist. So kann man Kollisionsabfragen pixelgenau durchführen, wenn man zwei Quellen miteinander verknüpft (dafür sind nämlich die zwei zusätzlichen Planes 6 & 7). Auch braucht man, um einen Speicherbereich zu löschen, keine Quellen.

Drösel: *Aha, aber ich hör' immer verknüpfen. Wie soll denn das gehen?*

Grünfried: Hierfür sind die unteren 8 Bits des BLTCON0 zuständig (Tab. 1).

Drösel: *Bleiben noch vier Bits übrig. Wofür?*

Grünfried: Mit ihnen kann ich (und/ oder Du und/oder Müllers Kuh) die Quelle A bitweise nach links verschieben - ohne Zeitverlust (realtime-shifter oder barrelshifter nennt sich das). Dafür sind die obersten 4 Bits zuständig. Allerdings wird links mit Nullen gefüllt und rechts abgeschnitten. Wenn Du also ein Raumschiff pixelweise verschieben willst, ohne daß ab und zu was fehlt, dann mußt du rechts sechzehn Pixel freilasssen.

Drösel: *Also das Raumschiff ein Word breiter machen, als es sein müßte.*

Grünfried: Ach nee - guck dir doch mal die Raumschiffdaten an. Aber damit Du's ganz genau weißt: Nehmen wir mal an, Du wolltest ein Taschentuch in 32 Farben über den Bildschirm hüpfen lassen. Dann bräuchtest Du eine Schattenmaske von dem Taschentuch (eine Plane, die sich ergibt, wenn man alle fünf Planes "zusammenANDet"). Die bräuchtest Du, damit der Blitter nur Hintergrunddaten ändert, wo auch wirklich Taschentuch ist. Sie sei die Quelle A. Die Quelle B sei eine beliebige Plane der Grafikdaten. Die Quelle C sei die Umgebung, in die Du hineinblittest, damit ich auch weiß, was ich blitten

```
WAIT_JOYSTICK:                          ; auf Knopfdruck warten
  BTST     #7,CIAA_PRA                  ; Knopf gedrückt?
  BNE.S    WAIT_JOYSTICK                ; wenn nicht, wieder prüfen
  RTS
FAST_RAND:                              ; liefert Zufallszahl von 0-256
  MOVE.W   D1,-(A7)                     ; D1 auf den Stack retten
  MOVEQ    #0,D0                        ; D0 löschen
  MOVE.W   VHPOSR,D0                    ; Kathodenstrahlposition auslesen
  LSR.W    #8,D0                        ; Y-Wert nach unten schieben
  MOVE.W   VHPOSR,D1                    ; Kathodenstrahlposition auslesen
  EOR.B    D1,D0                        ; noch zufälliger machen
  MOVE.W   (A7)+,D1                     ; D1 vom Stack holen
  RTS
WAIT_BLT:                               ; wartet auf den Blitter
  MOVE.W   D0,-(A7)                     ; D0 auf den Stack retten
WB_LOOP:
  MOVE.W   DMACONR,D0                   ; DMACON-Register auslesen
  BTST     #14,D0                       ; Blitter am werken?
  BNE.S    WB_LOOP                      ; wenn versch. 0, dann nochmal
  MOVE.W   (A7)+,D0                     ; D0 vom Stack holen
  RTS
CALC_POS:                               ; Positionen umrechnen D0=x D1=y
  MOVE.W   D1,D2                        ; y-Position retten
  MOVE.W   D0,D1                        ; x-Position kopieren
  LSR.W    #4,D0                        ; Anzahl der Bytes auf der X-Achse
  LSL.W    #1,D0                        ; Anzahl der Worte auf der X-Achse
  ANDI.W   #$000f,D1                    ; Feinwert berechnen
  MULU     #FIELD_B,D2                  ; mal Bytes pro Zeile nehmen
  RTS
 [...]
  LEA      PERIODS,A2               ; Adresse holen
  MOVEA.L  (A2,D1.W),A1             ; Adresse der Periodtabelle holen
  LEA      AKT_PER,A2               ; Adresse der akt. Tabelle holen
  MOVE.L   A1,(A2,D6)               ; Adresse der akt. Per.tab abspeichern
  LEA      WAVES,A2                 ; Adresse holen
  MOVEA.L  (A2,D1.W),A1             ; Adresse der Wave holen
 [...]
LENGTHS: DC.W      4                    ; Länge der Wellen in Words
PERIODS: DC.L      PER1                 ; Adressen der Periodtabellen
AKT_PER: DC.L      PER1,PER1            ; Adresse der aktuellen Per.tab.
         DC.L      PER1,PER1            ; sind hier abgelegt
DMA_OFF: DC.W      $0001,$0002          ; Werte des DMA-Registers
         DC.W      $0004,$0008          ; zum Ausschalten einer Stimme
 [...]
  DC.W     10,5,9,5,8,6,7,6,6,6,5,6,4,6,3,6,2,6
  DC.W     1,6,0,$ffff
PER1:
  DC.W     54727,51656,48756,46020 ; Periodwerte für C0 bis H0
  DC.W     43437,40999,38698,36526 ; -> für die Frequenzen von
 [...]
  DC.W     $405,$7,$30,$408,$7,$30,$407,$7,$30,$405,$0,$C0,$0
  DC.W     $180,$0,$180,$8000
; -------------------------------------------------------------------
;
; Ab hier stehen die Routinen zum Aufbau und zur Bewegung der Sterne
;
; -------------------------------------------------------------------
INIT_STARS:
  MOVE.W   #STARS-1,D2                  ; Anzahl der Sterne als Zähler
  LEA      STAR_DUMMY,A0                ; Leeradresse
  LEA      STARS_DEL,A1                 ; Adresse der Adressentabelle
  LEA      STARS_POS,A2                 ; Adresse der Positionstabelle
  LEA      STARS_SPEED,A3               ; Adresse der Geschwindigkeitstab.
IS_LOOP:
  MOVE.L   A0,(A1)+                     ; Löschadresse schreiben
  JSR      FAST_RAND                    ; Zufallszahl holen
  MOVE.W   D0,D1                        ; nach D1 kopieren
```

muß, wenn die Quelle A Null ist. Das Ziel ist das Ziel. In BLT0CON schreibe ich $0fca. Was heißt das?

Drösel: *Daß A nicht verschoben wird, daß A,B,C und Ziel benutzt werden, und daß D=C, wenn A=0 und D=B, wenn A=1.*

Grünfried: Jou. Aber stell Dir vor, A wäre geshiftet gewesen, dann hätte B auch geshiftet sein müssen, damit die Sache funktioniert. Lösung: BLTCON1 hat in den obersten vier Bits analog zu BLTCON0 einen Shifter für die Quelle B.

Sebastian: *(aus dem Hintergrund) Shift or Caps Lock?*

Grünfried & Drösel: (heftig) Grafiker halten sich da raus!

Drösel: *Und wie startet man das Dingens?*

Grünfried: (mit Mühe Fassung bewahrend) Moment. Wie ich die Adressen setze, dürfte auch klar sein: Longs in BLTxPTH schreiben. Und unser gutes, altes Modulo gibt es auch auf dem Blitter. Analog zum Bitplane-Modulo ist das Blitter-Modulo in Bytes (und natürlich für A, B, C & D) vorhanden. Weiterhin gibts beim Blitter noch die Möglichkeit, das erste und das letzte Wort in jeder Zeile zu maskieren, indem man eine AND-Maske in die Register BLTFWM fürs erste und BLTLWM fürs letzte Wort schreibt.

Drösel: *Ja. Das ist ja ganz toll, aber wie starte ich das Ding denn?*

Grünfried: Dazu komm' ich ja gerade: Wenn man die Breite und Höhe setzt, wird der Blitter gestartet.

Drösel: *Genial.*

Grünfried: Nä, aber es kommt noch besser: Die unteren 6 Bits geben die Anzahl der Worte pro Zeile an (also max. 1024 Pixel), und die restlichen 10 Bits geben die Höhe des Objekts an, das geblittet werden soll (also auch max. 1024 Lines).

Drösel: *Na, damit wär' der Blitter wohl abgehakt.*

```
 JSR       FAST_RAND               ; Zufallszahl holen
 LSR.W     #2,D0                   ; vierteln
 ADD.W     D1,D0                   ; die beiden Zahlen addieren
 MOVE.W    D0,(A2)+                ; Position abspeichern
 JSR       FAST_RAND               ; neue Zufallszahl nach D0 holen
 ANDI.B    #$07,D0                 ; Geschwindigkeit zwischen 0 & 7
 ADDQ.B    #2,D0                   ; Geschwindigkeit zwischen 2 & 9
 MOVE.W    D0,(A3)+                ; Geschwindigkeit abspeichern
 DBRA      D2,IS_LOOP              ; nächster Stern
 RTS
STAR_DUMMY:
 DC.L      0                       ; Hilfsadresse

DEL_STARS:
 MOVE.W    #STARS-1,D4             ; Anzahl der Sterne als Zähler
 LEA       STARS_ADR,A0            ; Adresse der 1. Adressentabelle
 LEA       STARS_DEL,A1            ; Adresse der 2. Adressentabelle
DS_LOOP:
 MOVE.L    (A1),A2                 ; Adresse der zu löschenden Stelle
 CLR.B     (A2)                    ; alten Stern löschen
 MOVE.L    (A0)+,(A1)+             ; Adresse kopieren
 DBRA      D4,DS_LOOP              ; nächsten Stern
 RTS

MOVE_STARS:
 MOVE.W    #STARS-1,D4          ; Anzahl der Sterne als Zähler
 LEA       STARS_ADR,A0         ; Adresse der Adressentabelle
 LEA       STARS_POS,A1         ; Adresse der Positionstabelle
 LEA       STARS_SPEED,A2       ; Adresse der Geschwindigkeitstab.
 MOVEA.L   FIELD_BASE1,A3       ; Adresse des unsichtb. Spielfeldes
 LEA       STARS_TAB,A4         ; Adresse der geshifteten Sterne
MS_LOOP:
 MOVE.W    (A1),D1              ; Position holen
 SUB.W     (A2)+,D1             ; Um Geschwindigkeit vermindern
 BPL.S     MS_NO_NEW_STAR       ; Ende der Zeile noch nicht erreicht?
 MOVE.W    #351,D1              ; neuer Stern am linken Rand
 JSR       FAST_RAND            ; neue Zufallszahl nach D0 holen
 ANDI.B    #$07,D0              ; Geschwindigkeit zwischen 0 & 7
 ADDQ.B    #2,D0                ; Geschwindigkeit zwischen 2 & 9
 MOVE.W    D0,-2(A2)            ; Geschwindigkeit abspeichern
MS_NO_NEW_STAR:
 MOVE.W    D1,(A1)+             ; neue Position abspeichern
 MOVE.W    D1,D2                ; Position kopieren
 LSR.W     #3,D1                ; Position in Bytes
 ANDI.W    #$0007,D2            ; Bytes ausblenden
 MOVE.B    0(A4,D2),D3          ; verschobenen Stern holen
 LEA       0(A3,D1),A6          ; Zieladresse im Spielfeld
 MOVE.B    D3,(A6)              ; Stern ausgeben
 MOVE.L    A6,(A0)+             ; Position abspeichern
 ADDA.L    #LINES*FIELD_B,A3    ; 4 Zeilen weiter gehen
 DBRA      D4,MS_LOOP           ; nächsten Stern
 RTS

STARS_TAB:
 DC.B      %10000000,%01000000,%00100000,%00010000
 DC.B      %00001000,%00000100,%00000010,%00000001



; -------------------------------------------------------------------------------
;
; Hier stehen die Routinen zum Löschen und Bewegen des Schiffs
;
; -------------------------------------------------------------------------------

DEL_SHIP:
 JSR       WAIT_BLT                ; auf Ende des Blits warten
 MOVE.W    #$0100,BLTCON0          ; D wird auf Null gesetzt
 MOVE.W    #$0000,BLTCON1          ; zweites Kontrollreg. setzen
 MOVE.W    #FIELD_B-SHIP_B,BLTDMOD ; Modulo setzen
 MOVE.L    SH_BE_LAST,D0           ; da war das Schiff vor 2 Bildern
 MOVEQ     #5,D1                 ; 6 Planes müssen bearbeitet werden
DSH_LOOP:
 JSR       WAIT_BLT                ; auf Ende des Blits warten
 MOVE.L    D0,BLTDPTH              ; Dort soll der Blitter löschen
 MOVE.W    #SH_BLIT,BLTSIZE        ; Blitter Starten
 ADDI.L    #FIELD_P,D0             ; eine Plane weiter gehen
 DBRA      D1,DSH_LOOP             ; nächster Blit
 MOVE.L    SH_LAST,SH_BE_LAST      ; vorletzte wird letzte
 RTS

MOVE_SHIP:
 JSR       JOY_MOVE                ; Joystick kontrolieren
```

Grünfried: Zum Glück! Jetzt können wir endlich die Routine zur Raumschiffbewegung besprechen. Ähnlich den Sternenroutinen ist sie in einen Teil zum Löschen und in einen Teil zum Hinblitten gesplittet.

Drösel: *Zum Löschen wird aber nicht der Blitter benutzt, oder?*

Grünfried: Aber wohl, mein Junge! Ist doch ideal: Setzt man den "Verknüpfcode" auf 0, so werden lauter Nullen geblittet, und wir können löschen. Nach diesem Prinzip funktioniert DEL_SHIP nämlich: Adresse holen - Blitter starten - warten - nächste Plane löschen. Aber ein wichtiges Element in der Kette fehlt uns: Drösel?

Drösel: *Ich hab' keine Lust mehr, in diesem saublöden Artikel die Antworten zu geben.*

Martin: Aber ich hab' dir doch meinen Schokoladenosterhasen versprochen!

Drösel: *Naja, dann: Wir wissen nicht, wie wir auf den Blitter warten sollen.*

Grünfried: Hierfür gibt's die Routine WAIT_BLT, die auf das BBUSY-Bit im DMACONR wartet (siehe dazu Kikkup2, S.58 - das ist die herbstliche KICKSTART, wo diese Tante guckt, als ob man sie gerade aufgeklärt hat).

Drösel: *Die Kickstart kenn' ich auch.*

Grünfried: Zur Sache bitte! Die Hauptroutine MOVE_SHIP funktioniert im Prinzip auch nicht anders, nur daß sie das Taschentuchprinzip anwendet.

Drösel: *Hä?*

Grünfried: Wir haben das oben besprochen, wie man was blittet, ohne, daß der Hintergrund 'puttgeht. Allerdings sei hier gesagt, daß, wenn der Hintergrund nicht jedesmal neu aufgebaut wird, das Stück, in das geblittet wird, vorher in einen Zwischenspeicher muß. Viel wichtiger jedoch ist die Routine JOY_MOVE, die die Koordinaten des Raumschiffes abhängig von einer Maus im Port #0 und einem Joystick im Port #1 ändert.

```
 MOVE.L   SH_X,D0                      ; D0 mit X-Position laden
 MOVE.L   SH_Y,D1                      ; D1 mit Y-Position laden
 JSR      CALC_POS                     ; Position umrechnen
 LSL.W    #8,D1                        ; Shiftwert um 8 Bits nach rechts
 LSL.W    #4,D1                        ; Shiftwert um 4 Bits nach rechts
 JSR      WAIT_BLT                     ; auf Ende des Blits warten
 MOVE.W   D1,BLTCON1                   ; zweites Kontrollreg. setzen
 ADDI.W   #$0fca,D1                    ; ABCD an und C=AB+Ac
 MOVE.W   D1,BLTCON0                   ; D wird auf Null gesetzt
 CLR.W    BLTAMOD                      ; Schattenmaske: kein Modulo
 CLR.W    BLTBMOD                      ; Schiffsdaten: kein Modulo
 MOVE.W   #FIELD_B-SHIP_B,BLTCMOD      ; Modulo setzen
 MOVE.W   #FIELD_B-SHIP_B,BLTDMOD      ; Modulo setzen
 ADD.L    FIELD_BASE1,D0               ; Adresse des Spielfelds zu D3
 ADD.L    D2,D0                        ; Y-Offset dazu
 MOVE.L   D0,SH_LAST                   ; vorletzte wird letzte
 MOVE.L   BOB_BASE,BLTBPTH             ; Zeiger auf Schiffsdaten setzen
 MOVE.L   BOB_BASE,D2                  ; Zeiger auf Daten
 ADDI.L   #5*SHIP_P,D2                 ; Adresse der Schattenmaske
 MOVEQ    #5,D1                      ; 6 Planes müssen bearbeitet werden
 MSH_LOOP:
 JSR      WAIT_BLT                   ; auf Ende des Blits warten
 MOVE.L   D2,BLTAPTH                 ; Zeiger auf Schattenmaske setzen
 MOVE.L   D0,BLTCPTH                 ; Dort liegt der Hintergrund
 MOVE.L   D0,BLTDPTH                 ; Dort soll der Blitter hinkopieren
 MOVE.W   #SH_BLIT,BLTSIZE           ; Blitter Starten
 ADDI.L   #FIELD_P,D0                ; eine Plane weiter gehen
 DBRA     D1,MSH_LOOP                ; nächster Blit
 RTS
JOY_MOVE:
 MOVE.W   JOY0DAT,D0                 ; Zähler auslesen
 MOVE.B   PREVX,D1                   ; vorherigen X-Wert holen
 MOVE.B   D0,PREVX                   ; jetzigen als vorigen abspeichern
 SUB.B    D0,D1                      ; jetzigen vom vorigen abziehen
 MOVE.B   D1,D2                      ; Wert abspeichern
 BPL.S    POSX                       ; Springe, wenn positiv
 NEG.B    D2                         ; positiv machen
POSX:     CMP.B    #127,D2           ; hat Zähler wieder bei Null angef.?
 BLE.S    NO_WRAPX                   ; springe, wenn nocht
 ADDI.B   #255,D1                    ; auf echten count bringen
NO_WRAPX:
 NEG.B    D1                         ; negieren
 EXT.W    D1                         ; auf Wort bringen
 EXT.L    D1                         ; auf Langwort bringen
 ADD.L    D1,SH_X                    ; neu setzen
MOUSE_Y:
 LSR.W    #8,D0                      ; shiften, um Y-Wert zu bekommen
 MOVE.B   PREVY,D1                   ; letzen Y-Wert holen
 MOVE.B   D0,PREVY                   ; jetzigen als vorigen abspeichern
 SUB.B    D0,D1                      ; jetzigen vom vorigen abziehen
 MOVE.B   D1,D2                      ; Wert abspeichern
 BPL.S    POSY                       ; springe, wenn größer Null
 NEG.B    D2                         ; positiv machen
POSY:     CMP.B    #127,D2           ; Umkehr des Zählers passiert
 BLE.S    NO_WRAPY                   ; springe, wenn nicht
 ADDI.B   #255,D1                    ; auf echten Wert bringen
NO_WRAPY:
 NEG.B    D1                         ; Vorzeichen wechseln
 EXT.W    D1                         ; auf Wort bringen
 EXT.L    D1                         ; auf Langwort bringen
 ADD.L    D1,SH_Y                    ; neu setzen
JOYSTICK_X:
 MOVE.W   JOY1DAT,D0                 ; Inhalt des Registers nach D0
 BTST     #1,D0                      ; Joystick nach rechts?
 BEQ.S    JM_NOT_RIGHT               ; springen, wenn nicht gedrückt
 ADDQ.L   #XSPEED,SH_X               ; Geschw. zur X-Position
 BTST     #0,D0                      ; Joystick nach hinten gedrückt?
 BNE.S    JOYSTICK_Y                 ; überspringen, wenn gesetzt
 ADDQ.L   #YSPEED,SH_Y               ; Geschw. zur Y-Position
JM_NOT_RIGHT:
 BTST     #0,D0                      ; Joystick nach hinten gedrückt?
 BEQ.S    JOYSTICK_Y                 ; überspringen, wenn nicht gesetzt
 ADDQ.L   #YSPEED,SH_Y               ; Geschw. zur Y-Position
JOYSTICK_Y:
 BTST     #9,D0                      ; Joystick nach links?
```

Drösel: *(springt energisch auf, zieht eine Pistole) Jetzt reicht's mir aber. Immer hast du recht, immer bin ich der Dumme. Da hast du, was du verdienst (schießt auf Grünfried: peng peng).*

Grünfried: Arghhh .... röchel .... (zensiert)

Alles Banane

Tja, das Experiment ging wohl daneben. Dann werd' ich den Maustreiber halt selber erklären müssen: Nehmen wir mal an, Sie hätten eine Maus. Die würde im Port #0 stecken. In der Maus sind dann zwei kleine Scheiben mit Schlitzen drin und einer Lichtschranke davor. AGNUS zählt heimlich diese Impulse, die entstehen, wenn man die Maus bewegt, und legt sie im JOY0DAT-Register ab (15-8=Y & 7-0=X). Würde man sich einen Wert merken und nach einiger Zeit einen neuen holen, könnte man dann ja anhand der Differenz den zurückgelegten Weg berechnen, wenn, ja wenn da nicht die Begrenzung des Zählers auf 8 Bit wäre. Es gibt also einen Umschlagspunkt. Aber es ist günstiger, sich den Maustreiber einfach unter den Nagel zu reißen, als ihn zu verstehen. Ähnlich kompliziert ist der Joysticktreiber. Denn auch hier wieder eine Besonderheit. Links und rechts werden ganz normal über Bit#1 und Bit#9 des JOYx-DAT abgefragt. Aber nach oben geht's nur, wenn Bit#9=1. Es sei denn, Bit#8 ist auch 1, dann geht's nur nach oben, wenn Bit#9=0. Analog Bit#1 und Bit#0 für unten.

Was fehlt jetzt eigentlich noch? Ich glaub', so unsystematisch hatten wir's noch nie, oder? Ach ja: Daß die Hauptschleife vom VERTB_M abhängig ist, dürfte jedem klar sein. Ebenso die WAIT_JOYSTICK-Routine und die COPY_COLORS. Und daß das Hauptspiel einen PORTS_M-Interrupt bekommt, der keine Musik spielt, liegt daran, daß die Musikroutine furchtbar viel Zeit braucht.

So, jetzt das BUFFER_READ: Beim Teil 2 hab' ich mal gesagt, daß ein Speeder, der mit einem Buffer arbeitet, um nicht jedes Byte einzeln laden zu müssen, mindestens 200 Zeilen bräuchte. Ich hab' das natürlich nur so Pi mal Daumen gesagt, weil ich meine Bilder

```
  BEQ.S    JM_NOT_LEFT          ; springen, wenn nicht gedrückt
  SUBQ.L   #XSPEED,SH_X         ; Geschw. von der X-Position abz.
  BTST     #8,D0                ; Joystick nach vorne gedrückt?
  BNE.S    TST_X                ; überspringen, wenn gesetzt
  SUBQ.L   #YSPEED,SH_Y         ; Geschw. zur Y-Position
JM_NOT_LEFT:
  BTST     #8,D0                ; Joystick nach vorne gedrückt
  BEQ.S    TST_X                ; überspringen, wenn nicht gesetzt
  SUBQ.L   #YSPEED,SH_Y         ; Geschw. zur Y-Position
TST_X:
  MOVE.L   SH_X,D0              ; X-Koordinate nach D0
  BPL.S    NOT_NEGX             ; ist der wert kleiner 0?
  CLR.L    SH_X                 ; auf null setzen
NOT_NEGX:
  CMP.L    #MAXX,D0             ; groesser als Breite des Felds?
  BLE.S    TST_Y                ; wenn nicht, dann springen
  MOVE.L   #MAXX,SH_X           ; auf alten Wert zuruecksetzen
TST_Y:
  MOVE.L   SH_Y,D0              ; Y-Koordinate nach D0
  BPL.S    NOT_NEGY             ; ist der wert kleiner Null?
  CLR.L    SH_Y                 ; auf Null setzen
NOT_NEGY:
  CMP.L    #MAXY,D0             ; größer als Höhe des Felds?
  BLE.S    NOT_MOREY            ; wenn nicht, dann...
  MOVE.L   #MAXY,SH_Y           ; auf alten Wert zurücksetzen
NOT_MOREY:
  RTS                           ; zurück

PREVX:          DC.B   0
PREVY:          DC.B   0
SH_LAST: DC.L   0               ; letze Adresse des Raumschiffs
SH_BE_LAST:     DC.L   0        ; vorletze Adresse des Schiffs
SH_X:           DC.L   0        ; Horizontale Position des Schiffs
SH_Y:           DC.L   0        ; Vertikale Position des Schiffs

; ---------------------------
;  Ab hier stehen die
;  Grafikdaten für das Schiff
;  und den Schuß
; ---------------------------

SHIP_DATA:
  dc.w     $f800,$0000,$0000
  dc.w     $fc00,$0000,$0000
  dc.w     $fe00,$0000,$0000
  dc.w     $1fe0,$0000,$0000
  dc.w     $e000,$0000,$0000
  dc.w     $ffff,$0000,$0000
  dc.w     $1ffd,$c000,$0000
  dc.w     $0005,$f000,$0000
  dc.w     $ffe0,$0fc0,$0000
  dc.w     $e004,$0002,$0000
  dc.w     $1fe0,$2004,$0000
  dc.w     $0004,$1ff8,$0000
  dc.w     $2538,$6000,$0000
  dc.w     $2722,$0000,$0000
  dc.w     $1130,$0000,$0000
  dc.w     $0fc0,$0000,$0000
  dc.w     $0800,$0000,$0000
  dc.w     $0000,$0000,$0000

  dc.w     $4400,$0000,$0000
  dc.w     $7e00,$0000,$0000
  dc.w     $4900,$0000,$0000
  dc.w     $e060,$0000,$0000
  dc.w     $0060,$0000,$0000
  dc.w     $e01f,$8000,$0000
  dc.w     $ffe6,$e000,$0000
  dc.w     $fffa,$3800,$0000
  dc.w     $e07f,$f860,$0000
  dc.w     $1fff,$0fff,$0000
  dc.w     $ffff,$f00e,$0000
  dc.w     $fffb,$e00c,$0000
  dc.w     $dac7,$fff0,$0000
  dc.w     $d8df,$c000,$0000
  dc.w     $7ff8,$0000,$0000
  dc.w     $7fe0,$0000,$0000
  dc.w     $3f80,$0000,$0000
  dc.w     $1e00,$0000,$0000

  dc.w     $0000,$0000,$0000
```

```
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000
        dc.w    $0000,$0000,$0000

        dc.w    $4000,$0000,$0000
        dc.w    $7c00,$0000,$0000
        dc.w    $4800,$0000,$0000
        dc.w    $0060,$0000,$0000
        dc.w    $0060,$0000,$0000
        dc.w    $e01f,$0000,$0000
        dc.w    $1fe5,$c000,$0000
        dc.w    $0019,$f000,$0000
        dc.w    $e06c,$0840,$0000
        dc.w    $1fe7,$0ffe,$0000
        dc.w    $1fe8,$300c,$0000
        dc.w    $001b,$c008,$0000
        dc.w    $dac7,$6000,$0000
        dc.w    $58de,$0000,$0000
        dc.w    $7ff0,$0000,$0000
        dc.w    $3fc0,$0000,$0000
        dc.w    $1f00,$0000,$0000
        dc.w    $0000,$0000,$0000

        dc.w    $fc00,$0000,$0000
        dc.w    $fe00,$0000,$0000
        dc.w    $ff00,$0000,$0000
        dc.w    $ffe0,$0000,$0000
        dc.w    $ffe0,$0000,$0000
        dc.w    $ffff,$8000,$0000
        dc.w    $ffff,$e000,$0000
        dc.w    $ffff,$f800,$0000
        dc.w    $ffff,$ffe0,$0000
        dc.w    $ffff,$ffff,$0000
        dc.w    $ffff,$fffe,$0000
        dc.w    $ffff,$fffc,$0000
        dc.w    $ffff,$fff0,$0000
        dc.w    $ffff,$c000,$0000
        dc.w    $7ff8,$0000,$0000
        dc.w    $7fe0,$0000,$0000
        dc.w    $3f80,$0000,$0000
        dc.w    $1e00,$0000,$0000

        dc.w    $fc00,$0000,$0000
        dc.w    $fe00,$0000,$0000
        dc.w    $ff00,$0000,$0000
        dc.w    $ffe0,$0000,$0000
        dc.w    $ffe0,$0000,$0000
        dc.w    $ffff,$8000,$0000
        dc.w    $ffff,$e000,$0000
        dc.w    $ffff,$f800,$0000
        dc.w    $ffff,$ffe0,$0000
        dc.w    $ffff,$ffff,$0000
        dc.w    $ffff,$fffe,$0000
        dc.w    $ffff,$fffc,$0000
        dc.w    $ffff,$fff0,$0000
        dc.w    $ffff,$c000,$0000
        dc.w    $7ff8,$0000,$0000
        dc.w    $7fe0,$0000,$0000
        dc.w    $3f80,$0000,$0000
        dc.w    $1e00,$0000,$0000

SHOOT_DATA:
        dc.w    $d3d3,$0000
        dc.w    $e1e1,$0000
        dc.w    $fbfb,$0000
        dc.w    $fbfb,$0000
        dc.w    $0404,$0000
        dc.w    $ffff,$0000
```

Listing geht weiter...

Bit:	*Name:*	*Beschreibung*
15	ASH3	Shiftwert für Quelle A Bit #3
14	ASH2	Shiftwert für Quelle A Bit #2
13	ASH1	Shiftwert für Quelle A Bit #1
12	ASH0	Shiftwert für Quelle A Bit #0
11	USEA	Quelle A an/aus
10	USEB	Quelle B an/aus
9	USEC	Quelle C an/aus
8	USED	Ziel an/aus
7	LF7	Wert für D wenn A=1, B=1 & C=1
6	LF6	Wert für D wenn A=1, B=1 & C=0
5	LF5	Wert für D wenn A=1, B=0 & C=1
4	LF4	Wert für D wenn A=1, B=0 & C=0
3	LF3	Wert für D wenn A=0, B=1 & C=1
2	LF2	Wert für D wenn A=0, B=1 & C=0
1	LF1	Wert für D wenn A=0, B=0 & C=1
0	LF0	Wert für D wenn A=0, B=0 & C=0

Tab.1: Das BLTCON0-Register

Bit:	*Name:*	*Beschreibung:*
15	BSH3	Shiftwert für Quelle A Bit #3
14	BSH2	Shiftwert für Quelle A Bit#2
13	BSH1	Shiftwert für Quelle A Bit#1
12	BSH0	Shiftwert für Quelle A Bit#0
11-2	nix	nix
1	DESC	Blitter zählt rückwärts, wenn 1
0	nix	nix

Tab. 2: Das BLTCON1-Register

immer gleich in mein Format bringe, und deshalb mich nie so recht mit IFF-loadern und DOS-Routinen beschäftigt hab'. Jedenfalls hab' ich sofort einen Brief auf Magnetlappen (Disk) bekommen, in dem mir ein freundlicher Mensch (der ausdrücklich ungenannt bleiben wollte)(aber er soll mir bitte seine neue Telefonnummer schicken) schrieb, sowas gäbe nie 200 Zeilen und belegte das mit einer eigenen Routine. An der hab' ich dann noch 'rumgebastelt und hab' sie dann eingebaut. Sie funktioniert zwar, aber sie ist kein wirklicher Ersatz! Um wirklich alle DOS-Funktionen zu erfüllen, bräuchte man mindestens 250 Zeilen. Aber was soll's. Mit genügend Buffer ist die Routine sauschnell.

Zum Schluß (wenn ich mich da nicht täusche) fehlt jetzt nur noch die Schußroutine. Nun ja, was soll ich viel dazu schreiben... Ich könnte Sie eigentlich nur verwirren. Praktisch ist es eine Mischung aus der STAR-Routine und der SHIP-Routine. Am besten ist es echt, wenn Sie sich den Source-Code anschauen.

Natürlich klingt's jetzt blöd, wenn ich sag', daß es schon wieder 5.10 Uhr in der Früh ist, und daß ich eigentlich schon um 20.00 Uhr hundemüde war... und glauben tut's mir eh keiner. Aber es stimmt - fragt meine Floppy. Und mal ganz unter uns Pfarrerstöchtern: sind Sie nicht auch froh, daß hier das ENDE ist?

P.S. Eben hat Fellini angerufen, er möchte mein Drehbuch von Grünfried und Drösel kaufen, aber ich werde wahrscheinlich ablehnen, weil mir die KICKSTART mehr Ruhm bieten kann.
P.P.S. Wenn Sie das hier nicht lesen, hab´ich doch An Fellini verkauft.

ENDE

VON SEBASTIAN DOSCH

KICKUP

Colorcycling und Materialeffekte

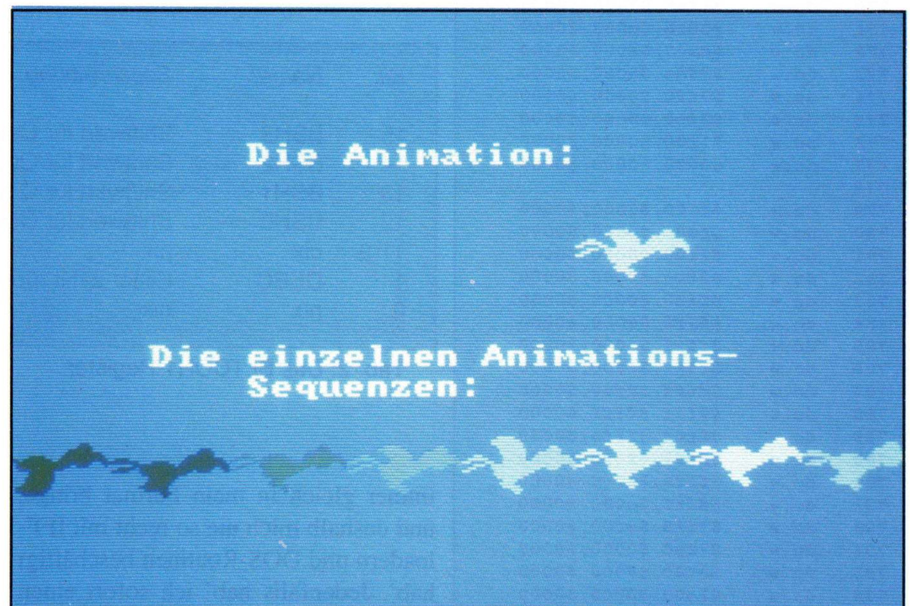
Was wäre ein Spiel ohne zündenden Vorspann? Dazu will ich Ihnen einige wichtige Tips geben, beispielsweise wie man Animationen und Materialeffekte mit DPaint kreieren kann. Dabei möchte ich Ihnen das Arbeiten mit DPaint etwas näherbringen und die Funktionen erklären.

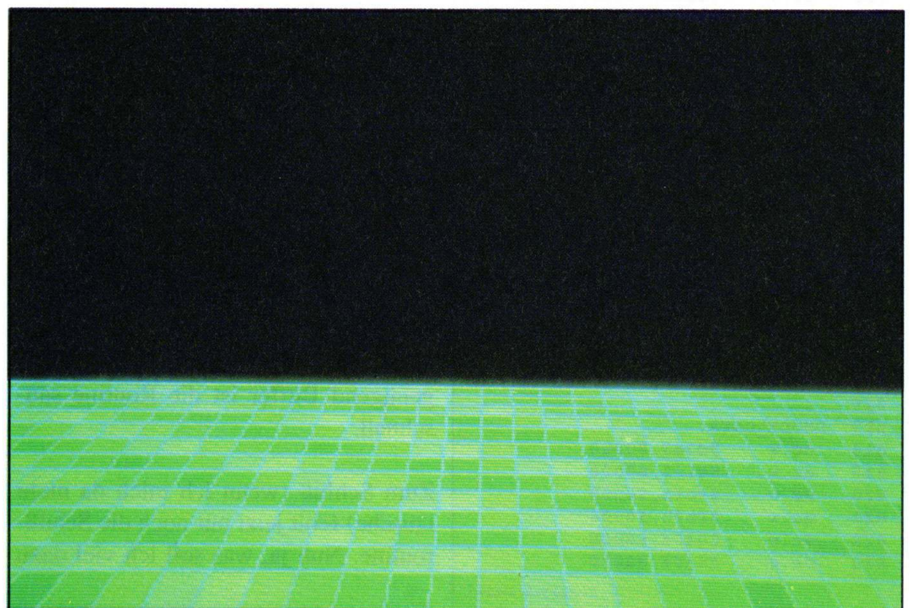
Animation in Deluxe Paint II

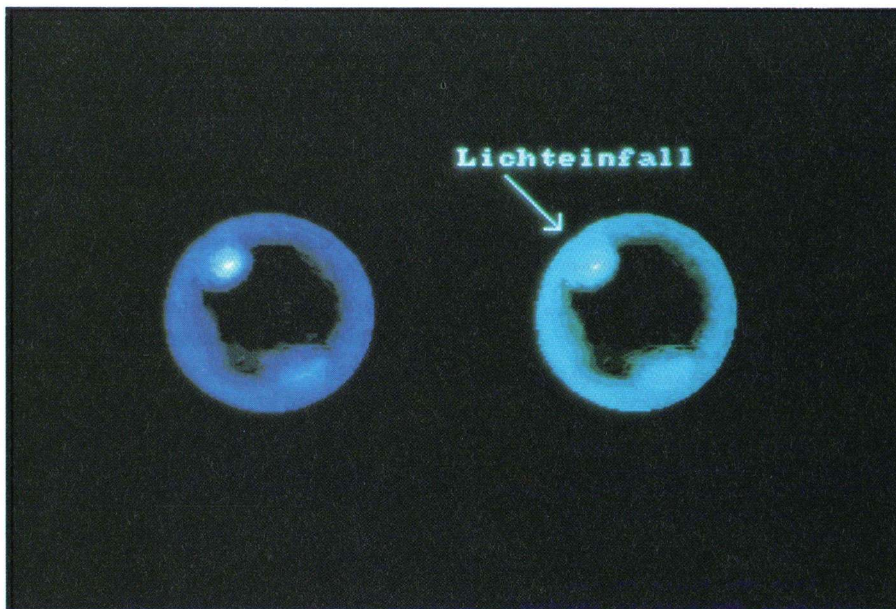
Ja, Sie haben ganz richtig gelesen, so unglaublich es auch klingen mag, es gibt doch eine Möglichkeit, Animationen in DPaint zu erstellen. Nein, dies ist keine eigenständige Funktion - Sie können die Anleitung ruhig wieder beiseite legen - es ist vielmehr eine Anwendung des Cycle-Befehles.
Dieser befiehlt Ihrem AMIGA, einen Farbaustausch in einem bestimmten Bereich und mit ein paar bestimmten, definierten Farben durchzuführen.

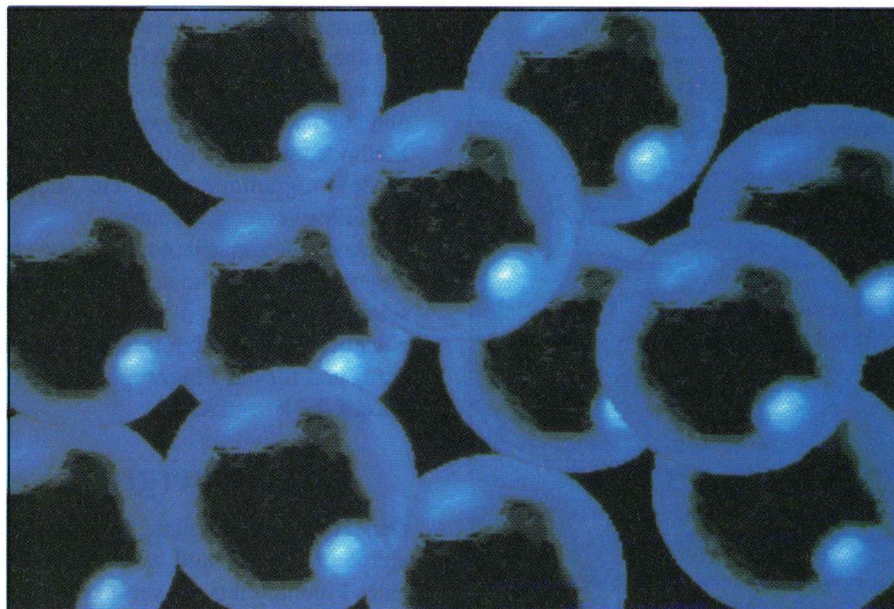
Der Cycle-Befehl kann in dem Color-Control-Menü unter dem Hauptverzeichnis 'Picture' ein- und ausgeschaltet werden; das Betätigen der Tabulatortaste bewirkt im übrigen dasselbe.

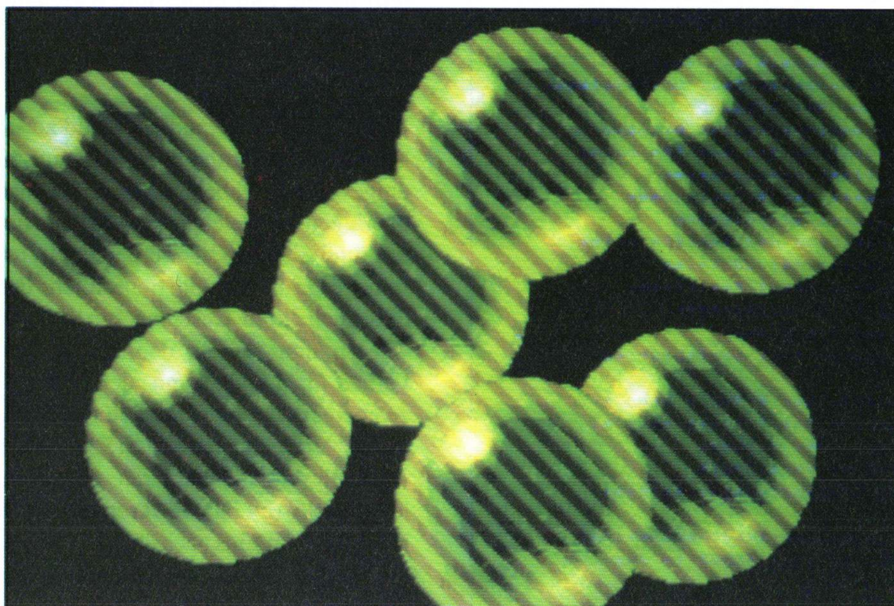
Das Rotieren von Farben beruht darauf, daß bestimmte Farben in einer Richtung nacheinander ausgetauscht werden. Gehen wir beispielsweise von der Palette *'a, b, c, d, e, f'* aus. Wir definieren einen Range von der Farbe *a* bis zur Farbe *f*, dieser bestimmt die Menge der Farben, die gecycled (rotiert) werden soll. Die Richtung der Farbrotation wird mittels des Pfeils im Palette-Window angegeben, die Geschwindigkeit mit dem Schieberegler über der UNDO-Funktion. Natürlich sollte man nicht vergessen, das Color-Cycling einzuschalten.
Nun wird im ersten Schritt die Farbe *a*











zur Farbe *b*, die Farbe *b* zur Farbe *c*, die Farbe *c* zur Farbe *d* und die Farbe *f* wieder zur Farbe *a*. Im zweiten und den darauf folgenden Schritten wiederholt sich das Ganze, jetzt nur mit den neuen Farben. Es läuft solange, bis der Cycle-Modus wieder deaktiviert wird.

Aber klar doch!!!

Aber klar doch - was? Dieser Aufruf sollte sich auf die erste Frage beziehen, in der gefragt wurde, ob es möglich sei, mit DPaint Animationen zu erzeugen. In der Tat, es ist nicht schwer. Sie können so beispielsweise ein Flugzeug über einen bewegten Hintergrund fliegen lassen oder Räder zum Drehen bringen, Menschen über den Bildschirm bewegen, Vögel aus der Tiefe des Bildschirmes nach vorne kommen lassen, Wasser zum Fließen bringen und vieles, vieles mehr.

Aber wie?

Ja, diese Frage ist berechtigt, doch die Lösung liegt nah. Auch hier tritt das Color-Cycling in Aktion. Nehmen wir als Beispiel den Flug eines Vogels, der sich von der linken zur rechten Bildschirmseite bewegt.
Zunächst werden die einzelnen Animationen, hier das Schlagen der Flügel, gezeichnet. Den Vogel belassen wir vorerst einfarbig, das ist einfacher, mit etwas Übung können Sie allerdings bis zu vier Farben für eine solche Animationssequenz verwenden. Die Anzahl der Animationssequenzen ist leider auf 31 begrenzt, doch das sollte genügen.

Nachdem Sie nun die einzelnen Bewegungsphasen auf dem Bildschirm dargestellt haben, müssen sie eingefärbt werden. Die erste Sequenz bekommt dabei die erste Vordergrundfarbe zugeordnet, die zweite Sequenz die zweite Farbe, die dritte die dritte, und so weiter, bis alle Animationssequenzen mit unterschiedlichen Farben gefüllt sind. Aber achten Sie bitte darauf, daß die Reihenfolge der Farben eingehalten wird.
Um die Animation zu gestalten, müssen wir uns erst mal überlegen, wie sie eigentlich aussehen soll. Ich schlage vor, der Vogel bildet eine schwarze Silhouette über hellblauem Hintergrund. Drükken Sie dazu die Taste 'P', worauf Sie prompt im Palette-Window landen. Färben Sie nun den Hintergrund, also die er-

... Fortsetzung

```
; -----------------------------------------------------------------
;
; Ab hier stehen die Routinen zur Verwaltung der Schüsse
;
; -----------------------------------------------------------------

DEL_SHOOTS:
  JSR     WAIT_BLT                  ; auf Ende des Blits warten
  MOVE.W  #$0100,BLTCON0            ; D wird auf Null gesetzt
  MOVE.W  #$0000,BLTCON1            ; zweites Kontrollreg. setzen
  MOVE.W  #FIELD_B-SHOOT_B,BLTDMOD ; Modulo setzen
  LEA     SO_BE_LAST,A0             ; Adresse der Adressen
  MOVE.W  SO_ALAST,D2               ; Anzahl der Schüsse am Schirm
  MOVE.W  AKT_SO,SO_ALAST           ; vorletzte wird letzte
DSO_LOOP2:
  SUBQ.W  #1,D2                     ; Schleifenzähler veringern
  BMI.S   DEL_SO_END                ; wenn neg, dann beenden
  MOVE.L  (A0)+,D0                  ; da war das Schiff vor 2 Bildern
  MOVEQ   #4,D1                     ; 5 Planes müssen bearbeitet
                                      werden
DSO_LOOP1:
  JSR     WAIT_BLT                  ; auf Ende des Blits warten
  MOVE.L  D0,BLTDPTH                ; Dort soll der Blitter löschen
  MOVE.W  #SO_BLIT,BLTSIZE          ; Blitter Starten
  ADDI.L  #FIELD_P,D0               ; eine Plane weiter gehen
  DBRA    D1,DSO_LOOP1            ; nächster Blit
  ADDI.L  #FIELD_P,D0             ; eine Plane weiter gehen
  JSR     WAIT_BLT                ; auf Ende des Blits warten
  MOVE.L  D0,BLTDPTH              ; Dort soll der Blitter löschen
  MOVE.W  #SO_BLIT,BLTSIZE        ; Blitter Starten
  BRA.S   DSO_LOOP2               ; nächster Schuß
DEL_SO_END:
  LEA     SO_LAST,A0              ; Adresse der Adressen
  LEA     SO_BE_LAST,A1           ; dito
  MOVE.W  AKT_SO,D0               ; Anzahl der aktiven Schüsse
  BEQ.S   DSO_END                 ; wenn Null, beenden
  SUBQ.W  #1,D0                   ; zum Schleifenzähler machen
DSO_LOOP3:
  MOVE.L  (A0)+,(A1)+             ; vorletzte wird letzte
  DBRA    D0,DSO_LOOP3            ; Schleife
DSO_END:
  RTS

MOVE_SHOOTS:
  TST.B   J_RELEASED              ; Knopf seit letztem Bild losgelassen?
  BEQ.S   TST_M_BUTTON            ; wenn nicht, dann überspringen
  BTST    #7,CIAA_PRA             ; Joystickknopf gedrückt?
  SF      J_RELEASED              ; Knopf-flag auf "gedrückt" setzen
  BEQ.S   MO_BUTTON               ; Wenn ja, dann überspringen
TST_M_BUTTON:
  TST.B   M_RELEASED              ; Knopf seit letztem Bild losgelassen?
  BEQ.S   MO_NO_BUTTON            ; wenn nicht, dann überspringen
  BTST    #6,CIAA_PRA             ; Knopf gedrückt?
  SF      M_RELEASED              ; Knopf-flag auf "gedrückt" setzen
  BNE.S   MO_NO_BUTTON            ; Wenn nicht, dann überspringen
MO_BUTTON:
  CMP.W   #SHOOT_M,AKT_SO         ; schon zuviele Schüsse?
  BEQ.S   MO_NO_BUTTON            ; Dann nächsten Teil überspringen
  MOVE.W  AKT_SO,D3               ; Anzahl der Schüsse
  LSL.W   #2,D3                   ; Auf Longwordzeiger bringen
  MOVE.L  SH_X,D0                 ; X-Position des Schiffs holen
  ADDI.L  #SHOOT_X,D0             ; X-Offset dazuaddieren
  LEA     SO_X,A0                 ; Adresse der X-Positionen
  MOVE.L  D0,0(A0,D3.W)           ; Position abspeichern
  MOVE.L  SH_Y,D0                 ; Y-Position des Schiffs holen
  ADDI.L  #SHOOT_Y,D0             ; Y-Offset dazuaddieren
  LEA     SO_Y,A1                 ; Adresse der Y-Positionen
  MOVE.L  D0,0(A1,D3.W)           ; Position abspeichern
  ADDQ.W  #1,AKT_SO               ; einen Schuß mehr
  BRA.S   LOADED                  ; nächsten Teil überspringen
MO_NO_BUTTON:
  LEA     SO_X,A0                 ; Adresse der X-Werte nach A0
  LEA     SO_Y,A1                 ; Adresse der Y-Werte nach A1
  MOVE.W  AKT_SO,D3               ; Anzahl der Schüsse
  SUBQ.W  #1,D3                   ; um Eins vermindern
  LSL.W   #2,D3                   ; Auf Longwordzeiger bringen
LOADED:
  LEA     SO_LAST,A2              ; Adresse der Zwischenspeicher
  MOVEQ   #0,D4                   ; Zähler, wenn Schuß im Aus
MSO_LOOP:
```

ste Palettenfarbe, hellblau, die zweite Farbe schwarz und die restlichen Farben allesamt wieder hellblau.

Aber hoppla - bis auf einen Vogel sind alle restlichen verschwunden. Das war ja auch Sinn und Zweck der Übung, denn erst so kann der Vogel zum Leben erweckt werden. Stellen Sie zunächst noch die Richtung und die Geschwindigkeit ein und verlassen dann das Palette-Menü.

Wenn Ihnen kein Fehler unterlaufen ist, können Sie nun durch sanfte Betätigung der TAB-Taste die Animation in Gang setzen. 'Flap .... Flap .... Flap ....', der Vogel flattert von links nach rechts über den Monitor.

Und es geht weiter ...

Kommen wir nun zu einem Flugzeug, das über eine Landschaft fliegt. Diesmal bewegt sich die Landschaft, sie rauscht unter dem Flugzeug hinweg. Das ist eine Methode, mit deren Hilfe Sie auch ohne große Programmiertricks ein simples 'Scrolling' erzeugen können.

Zeichnen Sie zunächst ein Flugzeug, das ungefähr in der Mitte des Bildschirmes schwebt, und färben den Himmel in einer beliebigen Farbe ein. Benutzen Sie dafür und für das Flugzeug alle Farben bis auf die letzten acht.

Der Hintergrund

Den Hintergrund teilen Sie zuerst in einige Vierecke, und zwar mit Hilfe eines Fluchtpunkts, da sonst keine Tiefenwirkung entsteht. Zum Thema Tiefenwirkung (hat nichts mit Waschmitteln zu tun) können Sie in der Zeichenkiste nachlesen.

Wenn Sie nun, wie auf dem Bild zu sehen, den Untergrund in ein Karomuster umgewandelt haben, können Sie sich überlegen, in welchen Farben der Hintergrund dargestellt werden soll. Ich schlage grün vor und freue mich, daß dies auch Ihren Geschmack trifft. Wir werden nun also in den letzten acht Farben, beginnend mit der vierundzwanzigsten, einige Grüntöne anmischen. Die erste Farbe (Nr.24) färben wir hellgrün und kopieren sie an die letzte Stelle (Nr.31). Dort setzen wir die Helligkeit um Eins nach unten. Die mittlere Farbe (Nr.28) wird wunderschön dunkelgrün angemischt.

Nun tritt die Spread-Funktion in Aktion. Mit ihr verbinden wir die Farben 24 und 31. Als nächstes wird ein Range über die Farben 24 bis 31 gezogen, wonach die Farbpalette verlassen werden kann.

Bevor es an die Farbrotation geht, müssen die gezeichneten Rechtecke noch gefärbt werden. Beginnen Sie dabei links mit der ersten grünen Farbe, arbeiten sich nach rechts durch bis zur letzten Farbe und beginnen anschließend wieder mit der ersten.
Jetzt fehlt nur noch der Druck auf die Tab-Taste - ist doch hübsch, oder?

Was, noch mehr?

Ja, es gibt noch mehr Möglichkeiten, die ich Ihnen hier leider aus Platzgründen nicht mehr alle vorführen kann. Da wäre z.B. die Animation einer Kugel, die sich auf dem Bildschirm dreht oder fließendes Wasser, aufsteigende Kohlensäure in einem Mineralwasserglas, drehende Autoreifen oder Pfeile, die sich zur Verdeutlichung eines Schaubildes bewegen.
Wie Sie sehen ist dieses Thema unerschöpflich, und Sie benötigen hierzu keine besondere und vor allen Dingen keine teure Software.

Materialeffekte

Optisch interessant ist die Darstellung verschiedener Materialien. Beginnen wir mit einem sehr heiklen und zerbrechlichen Material: Glas.

Widmen wir uns zunächst Glas auf ebener Fläche und auf Kugeln. Für das Zeichnen von Glas sollten Sie sich einmal das Titelbild zu diesem Spiel in der Ausgabe "November '88" ansehen. An der Stelle, an der "Coding by..." steht, wird ganz eindrucksvoll gezeigt, wie solch ein Glaseffekt aussehen kann.

Wir nehmen uns zuerst eine Fläche, setzen falls nötig, einige Objekte oder Schriften darauf und füllen die Fläche dunkelgraublau aus. Dann wählen wir diese Farbe durch Klick mit der rechten Maustaste in der Palette als Brush aus.

Wir drücken nun die Taste 'F2' für Color, so daß der Brush eingefärbt wird, wählen die nächstdunklere Farbe zu der ursprünglichen Hintergrundfarbe und setzen den Brush, etwas nach rechts und

```
 TST.W   D3                    ; Schleifenzähler testen
 BMI     MO_END                ; Beenden, wenn negativ

 CMP.L   #MAXX_SO,0(A0,D3.W)   ; mit dem Maximalwert vergleichen
 BLE.S   SHOOT_X_OK            ; wenn kleiner oder gleich, dann OK
 SUBQ.W  #1,AKT_SO             ; ein Stern weniger
 BEQ     MO_END                ; beenden, wenn kein Schuß mehr
 MOVE.W  D3,D5                 ; Zeiger kopieren
 MOVE.W  D4,D6                 ; Zähler kopieren
MSO_COPY_LOOP:
 SUBQ.W  #1,D6                 ; Zähler veringern
 BMI     MO_NEXT               ; wenn negativ, dann überspringen
 MOVE.L  4(A0,D5),0(A0,D5)     ; X-Position kopieren
 MOVE.L  4(A1,D5),0(A1,D5)     ; Y-Position kopieren
 MOVE.L  4(A2,D5),0(A2,D5)     ; Zwischenspeicher kopieren
 ADDQ.L  #4,D5                 ; Zeiger erhöhen
 BRA.S   MSO_COPY_LOOP         ; Schleife
SHOOT_X_OK:

 MOVE.L  0(A0,D3.W),D0         ; D0 mit X-Position laden
 MOVE.L  0(A1,D3.W),D1         ; D1 mit Y-Position laden
 JSR     CALC_POS              ; Position umrechnen

 LSL.W   #8,D1                 ; Shiftwert um 8 Bits nach rechts
 LSL.W   #4,D1                 ; Shiftwert um 4 Bits nach rechts
 JSR     WAIT_BLT              ; auf Ende des Blits warten
 MOVE.W  D1,BLTCON1            ; zweites Kontrollreg. setzen
 ADDI.W  #$0fca,D1             ; ABCD an und C=AB+Ac
 MOVE.W  D1,BLTCON0                   ; D wird auf Null gesetzt

 CLR.W   BLTAMOD                      ; Schattenmaske: kein Modulo
 CLR.W   BLTBMOD                      ; Schussdaten:  kein Modulo
 MOVE.W  #FIELD_B-SHOOT_B,BLTCMOD     ; Modulo setzen
 MOVE.W  #FIELD_B-SHOOT_B,BLTDMOD     ; Modulo setzen
 ADD.L   FIELD_BASE1,D0               ; Adresse des Spielfelds zu D0
 ADD.L   D2,D0                        ; Y-Offset dazu
 MOVE.L  D0,(A2,D3)                ; vorletzte Adresse wird letzte

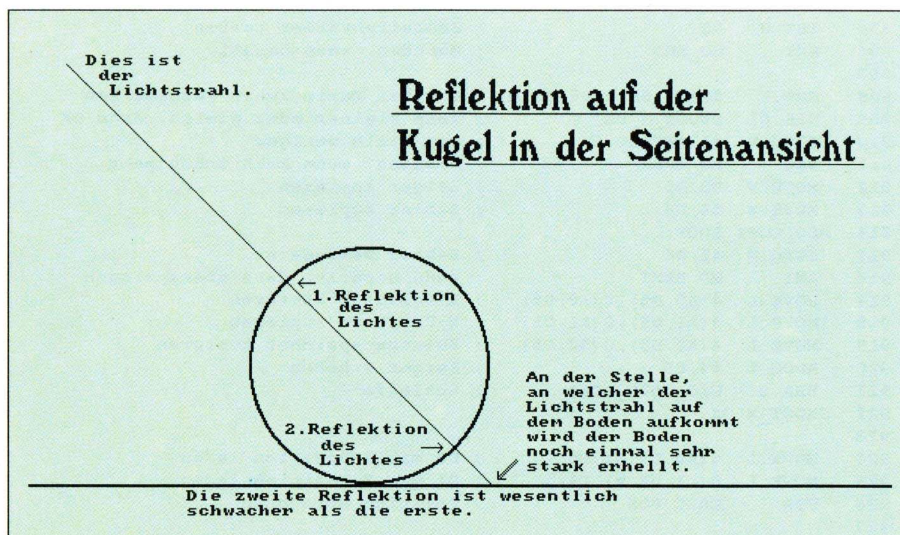
 MOVE.L  BOB_BASE,D2               ; Zeiger auf Schiffsdaten
 ADDI.L  #6*SHIP_P,D2              ; Adresse der Schussdaten berechnen
 MOVE.L  D2,BLTBPTH                ; Zeiger auf Schussdaten setzen
 ADDI.L  #5*SHOOT_P,D2             ; Adresse der Schattenmaske

 MOVEQ   #4,D1                     ; 5 Planes müssen bearbeitet werden
 MSO_LOOP2:
 JSR     WAIT_BLT                  ; auf Ende des Blits warten
 MOVE.L  D2,BLTAPTH                ; Zeiger auf Schattenmaske setzen
 MOVE.L  D0,BLTCPTH                ; Dort liegt der Hintergrund
 MOVE.L  D0,BLTDPTH                ; Dort soll der Blitter hinkopieren
 MOVE.W  #SO_BLIT,BLTSIZE          ; Blitter Starten
 ADDI.L  #FIELD_P,D0               ; eine Plane weiter gehen
 DBRA    D1,MSO_LOOP2              ; nächster Blit
 ADDI.L  #FIELD_P,D0               ; Adresse der Kollisions-Plane
 JSR     WAIT_BLT                  ; auf Ende des Blits warten
 MOVE.L  D2,BLTAPTH                ; Zeiger auf Schattenmaske setzen
 MOVE.L  D0,BLTCPTH                ; Dort liegt der Hintergrund
 MOVE.L  D0,BLTDPTH                ; Dort soll der Blitter hinkopieren
 MOVE.W  #SO_BLIT,BLTSIZE          ; Blitter Starten

 ADDI.L  #SHOOT_S,0(A0,D3.W)       ; Speed zur X-Position addieren
MO_NEXT:
 SUBQ.W  #4,D3                     ; Schleifenzähler veringern
 ADDQ.W  #1,D4                     ; einen Schuß mehr
 BRA     MSO_LOOP                  ; wenn neg, dann beenden
MO_END:
 RTS

TEST_BUTTONS:
 BTST    #7,CIAA_PRA               ; Joystickknopf gedrückt?
 SNE     J_RELEASED                ; Wenn ja, dann setzen
 BTST    #6,CIAA_PRA               ; Mausknopf gedrückt?
 SNE     M_RELEASED                ; Wenn ja, dann setzen
 RTS

 SO_X            DS.L    SHOOT_M   ; X-Positionen der Schüsse
 SO_Y            DS.L    SHOOT_M   ; Y-Positionen der Schüsse
 SO_LAST: DS.L   SHOOT_M           ; Adresse vor einem Videobild
 SO_BE_LAST:     DS.L    SHOOT_M   ; Adresse vor zwei Videobildern
AKT_SO:          DC.W    0         ; Anzahl der Schüsse auf dem Feld
 SO_ALAST:       DC.W    0         ; Anzahl vor einem Frame
M_RELEASED:      DC.B    0         ; Flag für den Mausknopfzustand
J_RELEASED:      DC.B    0         ; Flag für den Joys.knopfzustand
```




```
; ------------------------------------------------------------
;
; Ab hier stehen der Stack, Variablen und Zwischenspeicher (Buffer)
[...]
INTUI_BASE:     DC.L    0       ; Adresse der intuition-library
DOS_BASE:       DC.L    0       ; Adresse der dos-library
BOB_BASE:       DC.L    0       ; Adresse des Objektspeichers
TITLE_BASE:     DC.L    0       ; Adresse des Titlebild-Speichers
FIELD_BASE:     DC.L    0       ; Adresse des Spielfeld-Speichers
FIELD_BASE1:    DC.L    0       ; Adresse des Spielfelds #1
FIELD_BASE2:    DC.L    0       ; Adresse des Spielfelds #2
FILE:           DC.L    0       ; Adresse des File-Handles
TITLE_CMAP:     DS.W    32      ; Platz fuer die Titelbildfarben
STARS_DEL:      DS.L    STARS   ; zu löschende Sterne
STARS_ADR:      DS.L    STARS   ; letze benutzte Adresse
STARS_POS       DS.W    STARS   ; X-Position des Sterns
STARS_SPEED:    DS.W    STARS   ; Geschwindigkeit des Sterns
ERRORN:         DC.W    0       ; Nummer der Fehlermeldung
REST:           DC.L    0       ; Anz. der noch nicht geles. Bytes
POSITION:       DC.L    0       ; aktuelle Position im Buffer
BUFFER:         DC.L    0       ; Adresse des Buffers
ALERTS:         DC.L    ALERT0,ALERT1    ; Adressen der Alert-Texte
        DC.L    ALERT2,ALERT3
        DC.L    ALERT4,ALERT5
        DC.L    ALERT6,ALERT7
        DC.L    ALERT8
MAIN_CMAP:      DC.W    $000,$444,$ef0,$f8c,$aac,$669,$337,$115
        DC.W    $8dc,$5aa,$387,$155,$baa,$967,$735,$514
        DC.W    $bbb,$888,$444,$ffa,$fb0,$f60,$b00,$900
        DC.W    $700,$000,$999,$666,$bba,$9a7,$674,$452
DOS_NAME:       DC.B    "dos.library",0
GFX_NAME:       DC.B    "gfx.library",0
[...]
ALERT8:         DC.B    0,16,16,"Fehler #8: kein Speicher für "
        DC.B    "Read-Buffer da!",0,0
      END
```

unten verschoben, wieder auf den Bildschirm. Nach Druck der Taste 'F1' - man erhält dadurch wieder den ursprünglichen Brush - wählen wir den ersten Hintergrund als Hintergrundfarbe und setzen den Brush an seinen alten Platz.

Die Kugel

Aber kommen wir doch jetzt zur Kugel. Wenn Sie im Besitz eines Raytracing-Programmes sind, sind Sie natürlich aus dem Schneider, denn das erzeugt Ihnen Kugeln nach Maß. Wir nehmen aber jetzt an, wir hätten keines, oder so ein ekliger kleiner, grüner Virus mit einer dicken roten Brille (hallo Marc) hätte soeben Ihre Diskette mit dem Raytracing-Programm dahingerafft. Auf der Kugel definieren wir einen Punkt, auf dem das Licht am stärksten eintrifft. Nehmen wir an, das Licht kommt von links oben; der hellste Punkt befindet sich daher auch auf der linken Seite der Kugel. Das Licht trifft in die Kugel ein, wir dort gebrochen und hinterläßt seine Spuren auf der gegenüberliegenden Seite des Eintrittspunktes. Auf dem Bild sehen Sie ein auf diesem Prinzip beruhendes Bild. Nehmen Sie nun die Maus und versuchen Sie es nachzuvollziehen.

ENDE

Vorschau

Im nächsten KICKUP geht es hoch her. So kommen z.B. haufenweise Feinde ins Spiel, die sich wild auf dem Bildschirm tummeln und unserem Raumflitzer an den Kragen wollen.

Im grafischen Teil stellen wir Ihnen eine Vielzahl von Sprites für dieses Spiel vor, die Sie als Angreifer oder als eigenes Raumschiff einsetzen können.

Doch was wohl kaum jemand für möglich gehalten hat wird geschehen. Das große KICKUP-Ballerspiel wird vollendet.



ABO





ABO

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte mit 60 Pf. frankieren

Postkarte

Heim Verlag
Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57



Einzelheft- u. Monatsdisketten Bestellung





Einzelheft- u.
Disketten Service

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Bitte mit 60 Pf. frankieren

Postkarte

Heim Verlag
Heidelberger Landstr. 194

6100 Darmstadt-Eberstadt
Telefon 0 61 51 / 5 60 57



Kontaktkarte





Kontaktkarte

Bitte Adresse der Firma, bei der Sie Informationen, oder etwas bestellen möchten, auf der rechten Seite eintragen → → → → → → → → → →

Absender
(Bitte deutlich schreiben)

Vorname/Name

Beruf

Straße/Nr.

PLZ/Ort

Telefon Vorwahl/Rufnummer

Bitte freimachen

Postkarte

Firma

Straße/Postfach

PLZ Ort

KICK START AMIGA ZEITSCHRIFT

Abonnement

Ja, bitte senden Sie mir die Amiga-Computer Fachzeitschrift ab ______
für mindestens 1 Jahr (11 Hefte) zum ermäßigsten Preis von jährlich DM 70,— frei Haus.
(Ausland: Nur gegen Scheck-Voreinsendung DM 90,— Normalpost.)
Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Name ______

Vorname ______

Straße/Nr. ______

PLZ ______ Ort ______

Gewünschte Zahlungsweise bitte ankreuzen

☐ Bequem und bargeldlos durch Bankeinzug

Konto-Nr. ______ BLZ ______

Institut ______ Ort ______

☐ Ein Verrechnungsscheck über DM ______ liegt bei.

Garantie:
Diese Bestellung kann ich schriftlich innerhalb einer Woche (rechtzeitige Absendung genügt) widerrufen. Dies bestätige ich durch meine 2. Unterschrift.

Datum ______ Unterschrift ______

Datum ______ Unterschrift ______



KICK START AMIGA ZEITSCHRIFT

ABO

KICK START AMIGA ZEITSCHRIFT

Einzelheft- u. Monatsdisketten Bestellung

KICKSTART können Sie direkt beim HEIM-VERLAG zum Einzelheft-Preis von DM 7,– (zuzüglich Gebühr für Porto und Verpackung) nachbestellen. Bearbeitung nur gegen beigefügten Scheck über den entsprechenden Betrag (keine Überweisung).

Jan.	Febr.	März	April	Mai	Juni	Juli	Aug.	Sept.	Okt.	Nov.	Dez.	
												1987 = DM
												1988 = DM
												1989 = DM

\+ Gebühr für Porto u. Verp.

☐ Scheck in Höhe zus. DM liegt bei

Disketten Service

Alle Programme, die in KICKSTART veröffentlicht wurden, sind auf Disketten erhältlich. Die Disketten enthalten die Programme von jeweils 2 KICKSTART-Ausgaben. Bestellen Sie durch ankreuzen die gewünschten Disketten

Preis je Diskette 19,— DM								
	Juli/Aug. 87	Sept./Okt. 87	Nov./Dez. 87	Jan./Febr. 88	März/April 88	Mai/Juni 88	Juli/Aug. 88	Sept./Okt. 88
	Nov./Dez. 88	Jan./Febr. 89	März/April 89	Mai/Juni 89	Juli/Aug. 89	Sept./Okt. 89	Nov./Dez. 89	Jan./Febr. 90

Lieferung: gegen beigefügten Scheck zuzügl. 5,– DM Versandkosten (unabhängig von bestellter Stückzahl)



KICK START AMIGA ZEITSCHRIFT

Einzelheft- u. Monatsdisketten Bestellung

KICK START AMIGA ZEITSCHRIFT

☐ Ich bitte um weitere Informationen

☐ Ich gebe folgende Bestellung auf

in Bezug auf Ihre Anzeige in Kickstart Heft ______ Seite ______

Menge	Produkt/Bestellnummer	DM	gesamt DM

Datum, Unterschrift (für Jugendliche unter 18 Jahren der Erziehungsberechtigte)



Abgesandt am:

KICK START AMIGA ZEITSCHRIFT

Firma:

Bemerkungen:



'TOP 12'
Mein Lieblingsspiel



'TOP 12'
Mein Lieblingsspiel

Wettbewerbsbedingungen siehe TOP 12 in diesem Heft

Bitte frei- machen

MAXON - Computer GmbH
Redaktion KICKSTART
'TOP 12'
Industriestraße 26

D-6236 Eschborn



Kurzmitteilung





Kurzmitteilung

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Telefon

Postkarte

Bitte freimachen

MAXON Computer GmbH
KICKSTART Redaktion
Industriestraße 26

6236 Eschborn



PD Bestellung





PD Bestellung

Absender
(Bitte deutlich schreiben)

Vorname/Name

Straße/Nr.

PLZ/Ort

Postkarte

Bitte freimachen

MAXON Computer GmbH
KICKSTART Redaktion
Industriestraße 26

6236 Eschborn

KICK START
AMIGA ZEITSCHRIFT

'TOP 12'
Mein Lieblingsspiel

KICK START

KICK START
AMIGA ZEITSCHRIFT

Meine Meinung

Zu dem Artikel __________ in Heft ____, Seite ____
hätte ich folgendes zu bemerken:

☐ Ich möchte Ihnen folgendes Programm zur Veröffentlichung anbieten: (Kurzbeschreibung, Sprache, Länge in Druckerseiten)

☐ Ich kann über folgendes Thema berichten: (Tips & Tricks am AMIGA, Hardware, Software, etc.)

☐ Ich möchte gerne Autor in der KICKSTART werden. Meine Fachgebiete: (z.B. LISP, Modula-2, Assembler...)

☐ Ich möchte, daß folgendes Public-Domain Programm in Ihre Sammlung aufgenommen wird.

☐ Sonstiges

Bei weiteren Angaben oder Fragen wenden Sie sich bitte schriftlich oder telefonisch an die Redaktion. Tel. 0 61 96/48 18 11



KICK START
AMIGA ZEITSCHRIFT

Kurzmitteilung

KICK START
AMIGA ZEITSCHRIFT

PUBLIC DOMAIN SERVICE

Ich bestelle folgende PD-Disketten:
(Siehe PD Service in dieser Ausgabe)

Zahlung erfolgt:

☐ per Scheck

☐ per Nachnahme

(nur Inland) zuzüglich DM 3,70 Nachnahmegebühr

Ab 5 Disketten Versandkosten frei

Je Diskette fügen Sie bitte einen Betrag von DM 8,– bei,
für Porto und Verpackung je Sendung DM 5,– (Ausland DM 10,–)

Datum Unterschrift



KICK START
AMIGA ZEITSCHRIFT

PD Bestellung

Datum



KICKS
FÜR INSIDER

BE COOL!

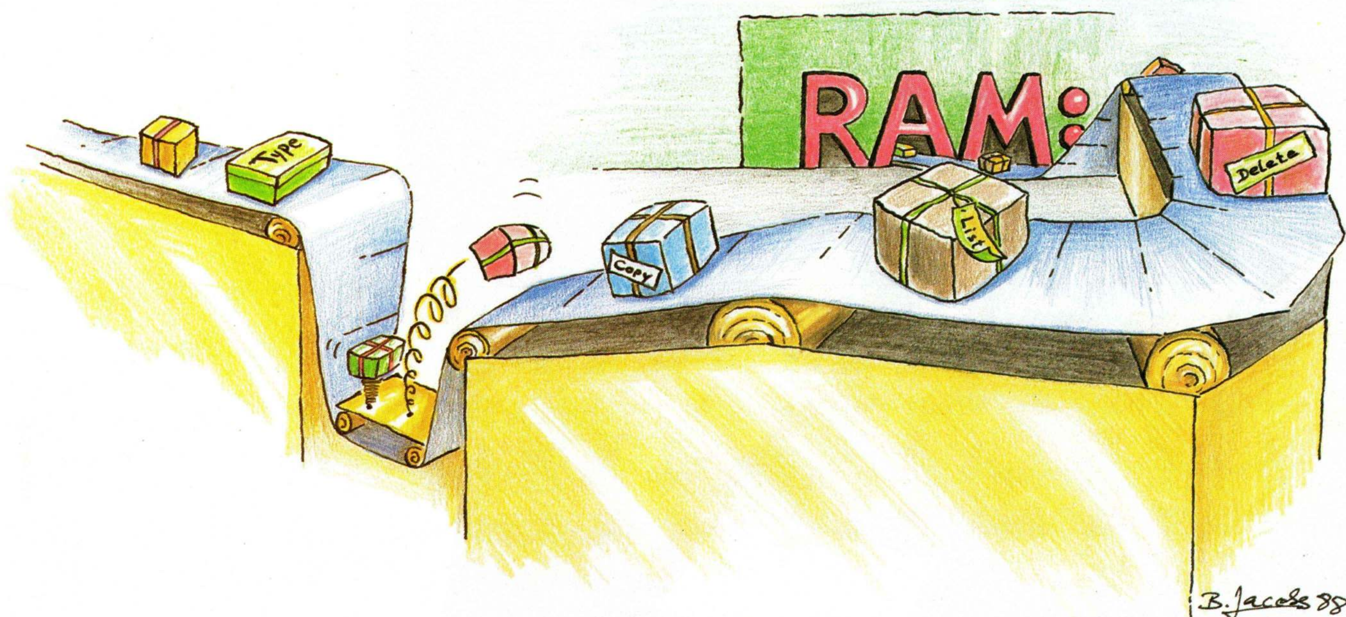
Beim Umgang mit dem Computer rauft man sich schon manchmal die Haare. Stürzt die Kiste doch nicht selten - ohne sichtbaren Grund - in die ewigen Abgründe. Computerviren geben ihren Teil hinzu. Wir empfehlen in diesem Fall, nicht in Panik auszubrechen. Bleiben Sie ruhig, überstürzen Sie nichts und schauen Sie sich die KICKS auf den folgenden Seiten an. Dort finden Sie unter anderem zwei Programme um den lästigen Viren entgegenzuwirken.

INHALT

VON TILO ROSSMANITH

MULTIPACK

Gut gepackt zur RAM-Disk



Das hier vorgestellte Programm nimmt Ihnen viel Arbeit ab. Wer kennt beispielsweise nicht das umständliche Kopieren von wichtigen Dateien auf die RAM-Disk? Man denke nur an die CLI-Befehle, die in den erzeugten C-Ordner der RAM-Disk kopiert werden sollen. Umständliches Kopieren und viele Eingaben sind vonnöten. Des weiteren braucht der Anwender schon eine gewisse Zeit, bis der Kopiervorgang abgeschlossen ist. Abhilfe schafft das hier vorgestellte Programm Multipack.

Mit MULTIPACK können Sie mehrere Dateien zu einer einzigen zusammenfügen und natürlich wieder zerlegen. Daß mit dem Programm eine Menge angestellt werden kann, zeigen die untenstehenden Beispiele. Besonders für Anwender, die oft die gleichen Dateien kopieren müssen, bietet sich der Einsatz geradezu an. Ein typisches Anwendungsbeispiel ist das Kopieren der CLI-Befehle auf die RAM-Disk. Jedoch möchte der Anwender nur eine begrenzte Zahl von Befehlen kopieren und nicht den ganzen C-Ordner. Das Programm sollten Sie in Ihren C-Ordner der Workbench-Diskette kopieren, dort ist es immer verfügbar. Daß das Programm nur vom CLI ausführbar ist, liegt am Programmierer, ein Starten von der Workbench ist wenig sinnvoll.

Die Syntax ist wie folgt festgelegt:

```
Multipack <GET|PUT|LIST|DEL> <Block>
                      [File_1 [..File_n]]
```

Die Funktionsweise des Programms läßt sich am besten an dem oben schon kurz erwähnten Beispiel verdeutlichen: Nehmen wir an, Sie möchten die drei Befehle *'dir'*, *'list'* und *'delete'* häufig auf eine RAM-Disk kopieren. Zu diesem Zweck erstellen Sie sich ein sogenanntes MULTIPACK-File, in dem die drei Files vereint werden.

Aufruf: (im CLI einzugeben)

```
Multipack PUT Commands.blk C:dir C:list
                          C:delete
```

Die PUT-Funktion bewirkt, daß ein MULTIPACK namens *'Commands.blk'* im aktuellen Directory geöffnet wird. In diesen MULTIPACK werden die drei Befehle hineinkopiert. Existiert bereits ein File mit dem Namen *'Commands.blk'*, so testet das Programm, ob dieses File ein Block ist. Ist dies nicht der Fall, wird mit einer Fehlermeldung zurückgesprungen. Nun fällt Ihnen ein, daß Sie auch den Befehl *'Type'* häufig verwenden weshalb Sie ihn an die MULTIPACK-Datei anhängen wollen:

```
Multipack PUT Commands.blk C:Type
```

Mit dem PUT-Befehl werden die bisherigen Files des MULTIPACKS nicht gelöscht.

Um diesen MULTIPACK nun auf die RAM-Disk zu bekommen, genügt ein weiterer Aufruf des MULTIPACK-Programms:

```
Multipack GET Commands.blk RAM:
```

Der GET-Befehl kopiert einen MULTIPACK in irgendein Verzeichnis und spaltet ihn in seine Einzelfiles auf. Weiterhin bietet das Programm die Möglichkeit auszugeben, welche Files in einem MULTIPACK beinhaltet sind:

```
Multipack LIST Commands.blk
```

Um einzelne Files aus einem MULTIPACK zu löschen, gibt es schließlich noch die Funktion DEL:

```
Multipack DEL Commands.blk list
```

Der obige Aufruf bewirkt, daß der List-Befehl aus dem MULTIPACK *'Commands.blk'* gelöscht wird.

Sie sehen, die Anwendung von MULTIPACK ist flexibel und nimmt Ihnen viel Arbeit ab. Ähnliches können Sie natürlich auch mit einer Batchdatei unternehmen, jedoch bedenken Sie, daß das Kopieren und Aufspalten der *Commands.blk*-Datei wesentlich schneller vonstatten geht als die Ausführung einer Batchdatei. Das daran, daß die Diskettenstruktur und die sich dabei aufbauende Dateistruktur eine komplexe Betriebssystemverwaltung voraussetzt. Mit MULTIPACK wird die Arbeit des Betriebssystems reduziert, die Folge ist eine wesentlich schnellere Abarbeitung, Sie werden überrascht sein. Für eine ca 100 Kbyte große *Command.blk*-Datei werden lediglich 5 Sekunden benötigt. Die gleiche Prozedur mit einer Batchdatei nimmt ca. 30 Sekunden in Anspruch.

MULTIPACK ist in der Porgrammiersprache C mit dem Aztec C-Compiler geschrieben. Zur Programmierung braucht nicht viel gesagt zu werden, die wichtigsten Programm-Module sind mit einem Kommentar versehen, so daß der Programmierer den Überblick behält.

```
/* Name: Multipack
   Autor: Tilo Rossmanith
   (c) MAXON Computer GmbH 1989
   KICKSTART 1989

  Dieses Programm ist in der Lage, mehrere
  Files in einem 'Block' unterzubringen,
  der dann schnell hin und her kopiert werden
  kann. Dies spart beim umkopieren z.B. auf
  die RAM-Disk eine Menge Zeit.

Syntax: 'Block <Mode> <Block> [<File1>..<FileN>]
  Mode: GET| PUT| LIST| DEL
         Bestimmt Arbeitsmodus
  Block: Filename des Blocks
  File: Filenamen, bei GET Zieldirectory
 */

 #include "exec/types.h"
 #include "libraries/dos.h"
 #include "exec/memory.h"

 #define BUFSIZE    10000

 /* ------------ OpenBlock ------------------ */

 long OpenBlock (name)
   char  *name;
 {
   long Fh;
   char test[4];

   Fh= Open (name, MODE_OLDFILE);
   if (Fh == NULL)
   {
     printf ("Block nicht gefunden\n");
     return (NULL);
   }

   Read (Fh, test, 4);
   if (strncmp ("FBLK", test, 4)!= NULL)
   {
     printf ("File ist kein Block\n");
     return (NULL);
   }

   return (Fh);
 }

 /* ----------- strcmpnc ----------------- */
 /* vergleicht 2 Strings ohne zw. Klein- und
    Grossbuchstaben zu unterscheiden */

 int strcmpnc (t1, t2)
   char *t1, *t2;
 {
   register char c1, c2;
   register COUNT count;

   for (count= 0; count<= strlen (t1); count++)
```

```
    {
        c1= t1[count];
        c2= t2[count];
        if (c1>= 65 && c1<= 90) c1+= 32;
        if (c2>= 65 && c2<= 90) c2+= 32;
        if (c1< c2) return (-1);
        if (c1> c2) return (1);
    }
    return (0);
}

/* -------------------- main -------------------- */

main(argc, argv)
    int argc;
    char *argv[];
{
    BYTE *Buffer= NULL;
    long SecondFh= NULL, BlockFh= NULL;
    long Size, Length;
    char *FileName;
    char Name[40], TextBuf[40];
    BYTE NameSize;
    COUNT Loop;
    char *Results[21];
    long lock= NULL;
    struct FileInfoBlock *fib= NULL;

    printf ("BLOCK - (C)09/88 Tilo Rossmanith\n");
    if (argc< 3)
        printf ("Syntax: 'Block <GET|PUT|LIST|DEL>
                <Block> [File_1..File_n]'\n");

    Buffer= (char *)AllocMem (BUFSIZE, NULL);
    fib= AllocMem (sizeof (struct FileInfoBlock),
            MEMF_CLEAR);
    if (Buffer== NULL || fib== NULL)
    {
        printf ("Kein Pufferspeicher verfuegbar\n");
        goto cleanup;
    }

/* ----- Files an Block anhängen ----- */

    if (strcmpnc (argv[1], "put")== NULL)
    {
        BlockFh= Open (argv[2], MODE_OLDFILE);
        if (BlockFh== NULL)
        {
            BlockFh= Open (argv[2], MODE_NEWFILE);
            if (BlockFh== NULL)
            {
                printf ("Block läßt sich nicht
                        erstellen\n");
                goto cleanup;
            }
            Write (BlockFh, "FBLK", 4);
        }
        else
        {
            Read (BlockFh, Name, 4);
            if (strncmp ("FBLK", Name, 4)!= NULL)
            {
                printf ("File '%s' ist kein Block\n",
                        argv[2]);
                goto cleanup;
            }
        }
        Seek (BlockFh, 0, OFFSET_END);

        for (Loop= 3; Loop< argc; Loop++)
        {
            lock= Lock (argv[Loop], ACCESS_READ);
            if (lock == NULL)
            {
                printf ("File '%s' nicht gefunden\n",
                        argv[Loop]);
                goto cleanup;
            }
            Examine (lock, fib);
            Size= fib-> fib_Size;
            UnLock (lock);
            FileName= argv[Loop];
            if (strchr (argv[Loop], ':')!= NULL)
                FileName= strchr (argv[Loop], ':')+1;
            while (strchr (FileName, '/')!= NULL)
                FileName= strchr (FileName, '/')+1;
            NameSize= strlen (FileName)+1;
            SecondFh= Open (argv[Loop], MODE_OLDFILE);

            Write (BlockFh, &Size, 4);
            Write (BlockFh, &NameSize, 1);
            Write (BlockFh, FileName, NameSize);

            do
            {
                Length=Read (SecondFh, Buffer, BUFSIZE);
                if (Write (BlockFh, Buffer, Length)!=
                        Length)
                {
                    printf ("Schreibfehler im Block\n");
                    printf ("ACHTUNG: Blockstruktur nun
                            vermutlich defekt\n");
                    Close (SecondFh);
                    goto cleanup;
                }
            } while (Length== BUFSIZE);

            Close (SecondFh);
        }
    }

/* --------- Block lesen --------- */

    if (strcmpnc (argv[1], "get")== NULL)
    {
        BlockFh= OpenBlock (argv[2]);
        if (BlockFh == NULL)
            goto cleanup;

        while (Read (BlockFh, &Size, 4)==4)
        {
            Read (BlockFh, &NameSize, 1);
            Read (BlockFh, Name, NameSize);

            TextBuf[0]= 0;
            strcat (TextBuf, argv[3]);
            if ((argv[3][strlen (argv[3])-1]!= ':')
                && (argv[3]!= NULL))
                strcat (TextBuf, "/");
            strcat (TextBuf, Name);

            SecondFh= Open (TextBuf, MODE_NEWFILE);
            if (SecondFh== NULL)
            {
                printf ("File '%s' läßt sich nicht
                        erstellen\n", TextBuf);
                goto cleanup;
            }

            do
            {
                Length= BUFSIZE;
                if (Size < BUFSIZE)
                    Length= Size;
                Read (BlockFh, Buffer, Length);
                if (Write (SecondFh, Buffer, Length)!=
                        Length)
                {
                    printf ("Schreibfehler bei File
                            '%s'\n", TextBuf);
                    Close (SecondFh);
                    DeleteFile (TextBuf);
                    goto cleanup;
                }
                Size-= Length;
            } while (Size> 0);

            Close (SecondFh);
        }
    }

/* ---------- Block auflisten ---------- */

    if (strcmpnc (argv[1], "list")== NULL)
    {
        BlockFh= OpenBlock (argv[2]);
```

```
    if (BlockFh == NULL)
      goto cleanup;

    while (Read (BlockFh, &Size, 4)==4)
    {
      Read (BlockFh, &NameSize, 1);
      Read (BlockFh, Name, NameSize);

      printf ("  %s\n", Name);

      Seek (BlockFh, Size, OFFSET_CURRENT);
    }
  }

 /* ------ Files aus Block loeschen ----- */

  if (strcmpnc (argv[1], "del")== NULL)
  {
    BlockFh= OpenBlock (argv[2]);
    if (BlockFh == NULL)
      goto cleanup;

    TextBuf[0]= 0;
    strcat (TextBuf, argv[2]);
    strcat (TextBuf, ".xtemp");

    SecondFh= Open (TextBuf, MODE_NEWFILE);
    if (SecondFh== NULL)
    {
      printf ("File '%s' läßt sich nicht
           öffnen\n", TextBuf);
      goto cleanup;
    }
    Write (SecondFh, "FBLK", 4);

    while (Read (BlockFh, &Size, 4)== 4)
    {
      Read (BlockFh, &NameSize, 1);
      Read (BlockFh, &Name, NameSize);
      for (Loop= 2; Loop< argc; Loop++)
      {
        if(strcmpnc (Name, argv[Loop])== NULL)
        {
          Loop= argc+10;
          Seek (BlockFh, Size, OFFSET_CURRENT);
        }
      }
      if (Loop< argc+10) /* umkopieren */
      {
        Write (SecondFh, &Size, 4);
        Write (SecondFh, &NameSize, 1);
        Write (SecondFh, Name, NameSize);

        do {
          Length= BUFSIZE;
          if (Length> Size)
            Length= Size;
          Read (BlockFh, Buffer, Length);
          if (Write (SecondFh, Buffer,
                Length)!= Length)
          {
            printf ("Schreibfehler in Block-
                File '%s'\n", Name);
            Close (SecondFh);
            DeleteFile (TextBuf);
            goto cleanup;
          }
          Size-= Length;
        } while (Size> 0);
      }
    }
    Close (SecondFh);
    Close (BlockFh); BlockFh= NULL;

    DeleteFile (argv[2]);
    Rename (TextBuf, argv[2]);
  }

 cleanup:

  if (Buffer)  FreeMem (Buffer, BUFSIZE);
  if (BlockFh)  Close (BlockFh);
  if (fib) FreeMem (fib, sizeof (
              struct FileInfoBlock));
 }
 /* LISTING ENDE */
```

END of KICK

KICKS FÜR INSIDER

Liebe Leser, KICKS für INSIDER ist eine Rubrik, die für interessante und lehrreiche Programme geschaffen wurde, denn was nutzt das schönste Programm, wenn man es nur abtippen, aber nicht verstehen kann.
Um jeden Monat neue KICKS zu veröffentlichen, sind wir natürlich auch von unseren Lesern abhängig, denn irgendwann einmal sind die Einfälle jedes Programmierers erschöpft. Also, nehmen Sie an dieser Rubrik aktiv teil, ergänzen Sie sie mit Ihren eigenen Programmen. Werden Sie Autor in der KICKSTART.
Dabei sind alle auf dem AMIGA existierenden Programmiersprachen gewünscht. Egal, ob Basic, C, Assembler, Modula 2, Pascal oder andere.

WIE DAS GEHT?

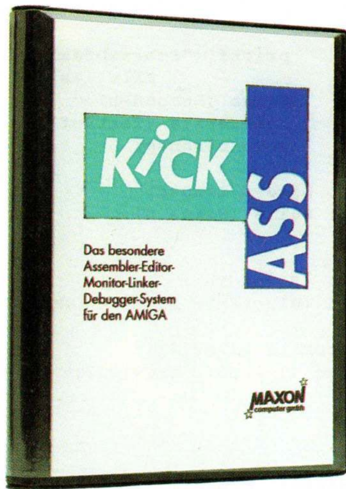
Ganz einfach. Schicken Sie uns das Programm, samt dem zugehörigen Artikel, eventuellen Bildern und Hardcopies (nach Möglichkeit "gegrabbt") auf Diskette zu. Wir werden uns dann bei Ihnen melden. Ein interessantes Honorar wartet bei Veröffentlichung natürlich auch auf Sie.
Wenn Sie Fragen haben, rufen Sie doch einfach bei uns an.

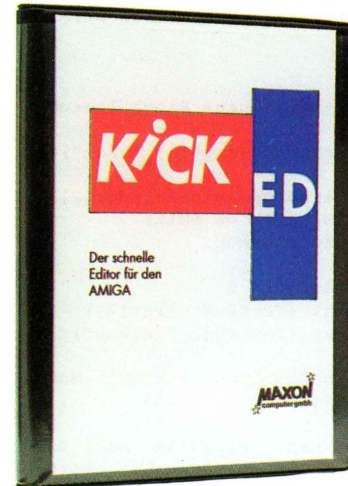
Tel. 06196/481811
Mo-Fr. 11 bis 17 Uhr

Ihre KICKSTART-Redaktion

KICK ME, AMIGA

Die Software mit dem gewissen KICK





KICK-ASS
Der besondere Assembler

KICK-ASS ist im Prinzip mehr als ein Assembler. Durch seinen integrierten EDITOR, MONITOR, DEBUGGER und LINKER ist KICK-ASS ein leistungsstarkes Entwicklungssystem, das die Entwicklung von Assembler-Programmen einfach, bequem und schnell macht. KICK-ASS wurde für all die entwickelt, die viel Wert auf Geschwindigkeit und Komfort legen.
Von den unzähligen Features, die KICK-ASS bietet, möchten wir nur einige nennen:

- Full-Screen-Editor mit vielen komfortablen Befehlen
- blitzschnelle Assemblierung schon während der Eingabe
- Der Quelltext wird stark komprimiert, so daß er sehr wenig Speicherplatz beansprucht.
- automatisches Einrücken der Zeilen, was gerade bei längeren Programmen eine hohe Lesbarkeit bewirkt
- automatische Syntaxüberprüfung schon bei der Eingabe
- Funktionen sind wahlweise durch Pull-Down-Menüs oder über Tastatur aufrufbar.
- Der integrierte LINKER fügt mehrere Programmodule zusammen. Sogar die Verbindung von Assembler mit Kompilaten von höheren Programmiersprachen ist möglich.
- Der DEBUGGER dient der Programmanalyse und Fehlersuche und ist eine unentbehrliche Hilfe beim Hineinschauen in fremde Programme.

KICK-ASS DM 89.-

KICK-ED
Der universelle Editor

KICK-ED können wir allen, die gern programmieren, sei es in 'C', Assembler, Pascal oder Modula-2, wärmstens empfehlen. Aber auch, 'um 'mal schnell' einen Brief zu schreiben und auszudrucken eignet sich dieser Editor.
KICK-ED ist ein professionelles Werkzeug für Programmierer. Die Bemühung um eine möglichst einleuchtende und angenehme Benutzerführung läßt sich an vielen Details aufzeigen:

- gleichzeitge Verwaltung mehrerer Texte, die mit Hilfe der Funktionstasten angewählt werden können
- Ein Hilfs-Menü, das jederzeit aufgerufen werden kann, gibt vollständige Information über die verschiedenen Befehle.
- bequemes Laden, Speichern und Anhängen von Texten über die integrierte Datei-Auswahl-Box
- Volle Multitasking-Unterstützung des AMIGA.
- superschnelles Softscrolling

KICK-ED DM 49.-

Bestellcoupon

Bitte senden Sie mir:

☐ KICK-ASS DM 89.-
☐ KICK-ED DM 49.-

zuzüglich Versandkosten:
Inland DM 7.50
Ausland DM 10.-

Name ____________
Straße ____________
Ort ____________
Datum ____________ Unterschrift ____________

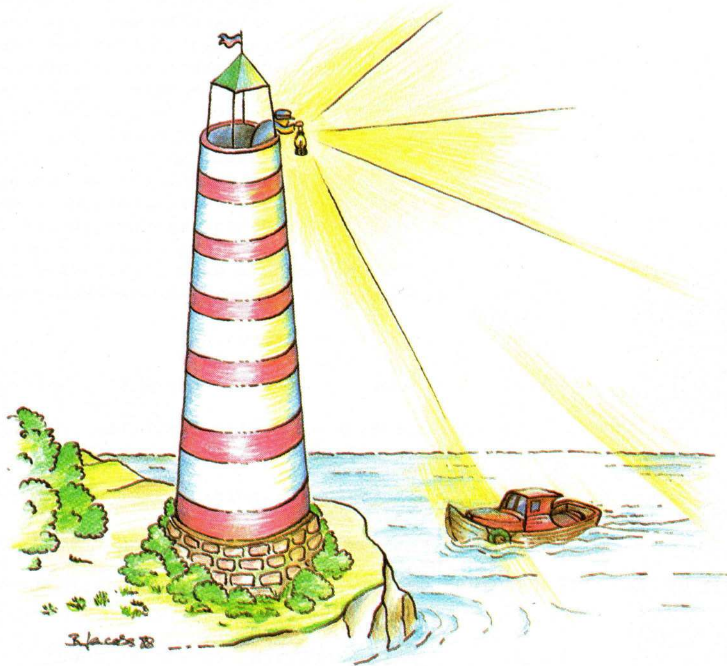
Auslandsbestellungen nur gegen Vorauskasse
Bei Nachnahme zuzüglich DM 3.50 Nachnahmegebühr

MAXON-Computer / Industriestr. 26 / 6236 Eschborn

VON TILO ROSSMANITH

Warnblinker

Virusmelder im Bootblock



Im Zeitalter der Computer-Viren kann der Anwender nicht vorsichtig genug sein, um seine wertvollen Programme und Daten vor einer Infizierung zu schützen. Mittlerweile gibt es die verschiedensten Antibiotika gegen die sehr nervenaufreibenden Viren. Doch der Schutz der Software ist es wert, immer neue Verfahren zu entwickeln.

Das hier abgedruckte Programm ist ein weiteres Medikament, um sich gegen den Befall zu schützen. Trotz seines simplen Aufbaus ist es sehr wirkungsvoll gegenüber Viren. Doch zunächst ein wenig Theorie über AMIGA-Viren: Die Viren beginnen ihr zerstörerisches Werk im Bootblock einer Diskette, von dort schreiben sie sich in den Speicher des AMIGA, normalerweise bemerkt der AMIGA-Anwender zunächst die Existenz der zumeist recht kleinen Programme nicht. Im Laufe der Zeit wundert sich der unbedarfte AMIGA-User über seltsame Dinge, die in seinem Computer vorgehen. Immer öfter erscheint die merkwürdige Meldung "SOMETHING WONDERFUL HAS HAPPENED" - ein Gag der AMIGA-Betriebssystem-Entwickler? Oder der Bildschirm wird auf einmal schwarz, und der AMIGA verweigert jegliche Weiterarbeit. Da hilft auch kein gutes Zureden, nur die Tastenkombination *linke Alternate-Taste - linke AMIGA-Taste - Space - rechte AMIGA-* und *rechte Alt-Taste* - und der AMIGA wird aus seinem Schlaf geweckt, allerdings nur für kurze Zeit. Dies sind nur die kleineren Übel. Aber wenn ausgerechnet das teure Lieblingsspiel von heute auf morgen nicht mehr zu laden ist, ist die Sache schon schwerwiegender. Alles Anzeichen, daß der AMIGA krank ist, oder vielmehr die Disketten mit Viren befallen sind. Um dem vorzubeugen, dient das kleine Programm BOOTBLINK, das im Boot-

block der Diskette seinen Platz findet. Es läßt den Bildschirm aufblinken, wenn das Programm Bootblink abgearbeitet ist. Blinkt der Bildschirm nicht auf, ist äußerste Vorsicht anzuraten; mit großer Wahrscheinlichkeit hat sich ein Virus eingenistet. Bei kopiergeschützter Original-Software sollte das Programm jedoch nicht installiert werden, da mitunter wichtige Daten im Bootblock stehen, die unter Umständen von unserem Programm überschrieben werden könnten.

Der Bootblock

In einem ganz normalen (Betriebssystem Version 1.2) Bootblock werden bekanntlich folgende Dinge angestellt:

```
start:
lea      dosname,a1 ; Adresse des
           Library-Namens nach A1
jsr      -96(a6)  ; FindResident
tst.l    d0       ; ist alles o.k.
beq      error    ; leider nicht
move.l   d0,a0
move.l   22(a0),a0
moveq    #$00,d0    ; ok
rts

error:
moveq    #$ff,d0    ; Fehler
rts

dosname:
'dos.library',0
```

Und jetzt kommt das Wesentliche: Wenn ich diesen Bootblock so abändere, daß er auch noch einen Bildschirm-Blinker eingebaut hat, erkenne ich sofort, ob eine Diskette meinen Spezial-Bootblock eingebaut hat oder nicht (bei einem anderen blinkt es nicht). Ich füge also an den Anfang des obigen Bootblocks einfach noch ein paar Zeilen an:

```
blink:
move.l  #$10000,d0
loop:
move.w  d0,$dff180; Hintergrund
                   Farbregister
sub.l   #$01,d0    ; dec. Zaehler
bne
loop              ; und Schleife
   .
   .              ; hier folgt der
   .              ; normale Bootblock
```

Und schon blinkt der Bildschirm kurz auf, sobald mein neuer Bootblock abgearbeitet wird.
Nun muß lediglich noch die Prüfsumme neu errechnet werden, und der Bootblock kann auf jede beliebige Diskette geschrieben werden.
So einfach ist es, einen sinnvollen Viren-Entdecker zu programmieren.

Das folgende Programm hat nun lediglich die Funktion, diesen Bootblock auf die Diskette zu schreiben. Es ist so einfach aufgebaut, daß sich sämtliche Erklärungen erübrigen.

```
/*====================================================
 Name          BootBlink
 Autor         Tilo Rossmanith
 Zweck         Aendert den Bootblock einer Diskette.
               Beim Booten der Disk blinkt nun der
               Bildschirm kurz auf, dadurch kann
               festgestellt werden, ob die Disk
               von einem Virus infiziert wurde
               (dann blinkt's nicht mehr!)
 Bedienung     Aufruf: 'BootBlink <drive>'
 Hinweise      Um das Programm kurz zu halten, wurde
               darauf verzichtet, alle moeglichen
               Fehlerquellen zu ueberpruefen. Es kann
               also vorkommen, dass das Programm keinen
               Fehler meldet, obwohl sich der Bootblock
               noch gar nicht auf der Disk befindet.
  (c) MAXON Computer GmbH ; KICKSTART 1989
=======================================================*/

#include "exec/types.h"
#include "exec/memory.h"
#include "devices/trackdisk.h"

struct MsgPort  *DiskPort= NULL;
struct IOStdReq *DiskRequest= NULL;

long    *BBlock= NULL;
long    BVal[17]= {
  0x444f5300, 0x4b8382e2, 0x00000370, 0x203c0001,
  0x000033c0, 0x00dff180, 0x538066f6, 0x43fa0018,
  0x4eaeffa0, 0x4a80670a, 0x20402068, 0x00167000,
  0x4e7570ff, 0x60fa646f, 0x732e6c69, 0x62726172,
  0x79000000
  /* Die kompletten Daten des neuen Bootblocks,
     der Rest wird mit Nullen aufgefuellt */
};

/* Universal Fehlerhandler */
void cleanup(text)
  char  *text;
{
  if (DiskRequest!= NULL)
  {
    /* Motor ausschalten */
    DiskRequest-> io_Command= TD_MOTOR;
    DiskRequest-> io_Length= 0;
    DoIO (DiskRequest);
    CloseDevice (DiskRequest);
    DeleteExtIO(DiskRequest,sizeof(struct IOStdReq));
  }
  if (DiskPort!= NULL)
    DeletePort (DiskPort);

  if (BBlock!= NULL)
    FreeMem (BBlock, 512*2);

  if (text)
    printf ("%s\n", text);

  exit ();
}

main(argc, argv)
  int  argc;
  char  *argv[];
{
  long  Unit;
  COUNT  Loop;
  long  Changenum;

```

```
    printf ("BootBlink - (C)10/88 MAXON Computer GmbH\n");

    Unit= argv[1][2]-'0';
    if ((argc!= 2) || (Unit<0 || Unit>3))
    {
      cleanup ("Aufruf: 'BootBlink <Drive>'\n");
    }

    /* Trackdisk-device oeffnen */
    DskPort= CreatePort (0,0);
    DiskRequest= CreateExtIO (DiskPort, sizeof (struct IOStdReq));
    if (OpenDevice (TD_NAME, Unit, DiskRequest, 0)!= 0)
    {
      cleanup ("Fehler beim Oeffnen des Devices\n");
    }

    /* Speicher fuer Bootblock */
    if ((BBlock= AllocMem (512*2, MEMF_CHIP|MEMF_CLEAR))== 0)
      cleanup("Kein Chip-Mem frei");

    /* Bootblock-Daten in den Diskpuffer umkopieren */
    for (Loop= 0; Loop< 17; Loop++)
      BBlock[Loop]= BVal[Loop];

    /* Bootblock auf Disk schreiben */
    DiskRequest-> io_Command= CMD_WRITE;
    DiskRequest-> io_Data= BBlock;
    DiskRequest-> io_Offset= 0; /* Block 0 */
    DiskRequest-> io_Length= 1024; /* 2 Blocks schreiben */
    DoIO (DiskRequest);

  /* Das folgende Kommando bewirkt, dass der Track-Puffer des
     trackdisk.device sofort auf die Disk geschrieben wird. */
    DiskRequest-> io_Command= CMD_UPDATE;
    DoIO (DiskRequest);
    if (DiskRequest-> io_Error!= NULL)
      cleanup ("Fehler beim Schreiben des Bootblocks");

    cleanup ("BootBlink wurde auf der Diskette installiert");
```

VON THOMAS WACKER

BLOCKBUSTER

Der Virenjäger hält die Augen auf



Neuer Virus im System? Kein Problem. Mit einem einfachen und sinnvollen Programm können alle Blöcke einer Diskette in hexadezimaler und ASCII-Form angezeigt werden. So kann man sich vergewissern, ob nicht vielleicht doch ein neuer Virus entstanden ist. Aber das Programm BLOCK kann noch viel mehr.

Schön und gut, aber man könnte doch einfach durch einen Diskettenmonitor nachsehen, ob die Diskette befallen ist. "Stimmt auch, aber nicht jeder besizt auch einen Diskettenmonitor. Und wenn er doch einen besizt, muß er erst unvorteilhaft die Disketten wechseln beziehungsweise ganz von neuem booten."

Das ist der eigentliche Vorteil dieses Programms. Es ist kurz, und fühlt sich am wohlsten im C-Ordner Ihrer Diskette. Es ist ein Muß für jeden, der sich für seine Diskettendaten interessiert. Aber nicht nur der Bootblock kann angesehen werden, auch alle 1758 anderen Blöcke! Die Ausgabe der Daten erfolgt geordnet in hexadezimaler und ASCII-Form.

Die Syntax des Befehls lautet:

BLOCK [dfx:[BlockNr [BlockNr BlockNr ...]

'*dfx*' steht für ein beliebiges Laufwerk, dies kann *df0:*, *df1:*, *df2:* oder *df3:* sein. '*BlockNr*' steht für den Block, den Sie sich betrachten wollen, wobei mehrere Blöcke auf einmal eingegeben werden können.

Beispiel:

BLOCK df1: 0

liest von der Diskette, die im Laufwerk *df1:* eingelegt ist, den Block Nummer 0.

BLOCK df0: 880 881 882

liest nacheinander die Blöcke 880, 881 und 882 ein. Die Ausgabe der Blöcke auf den Bildschirm kann mit CTRL-C unterbrochen werden. Durch Gedrückthalten des rechen Mausknopfes wird die Ausgabe der Blöcke angehalten, durch Loslassen wird sie fortgesetzt. Natürlich kann die Ausgabe auch durch einen

```
 1: /**************************************************/
 2: /* Programmname: Block                            */
 3: /* Anzeigen des Blockinhaltes                     */
 4: /* Autor: Thomas Wacker                           */
 5: /* (c) MAXON Computer GmbH 1989                   */
 6: /* KICKSTART 1989                                 */
 7: /**************************************************/
 8:
 9: #include <stdio.h>
10: #include <exec/memory.h>
11: #include <devices/trackdisk.h>
13: #include <functions.h>
14:
15: struct MsgPort          *dpt;
16: struct IOStdReq         *drq;
17: unsigned char           *dbuf;
18:
19: main(argc,argv)
20: int argc;
21: char *argv[];
22: {
23:  int i=1,j,n,blocknu,kk,fehler;
24:  long drive=NULL;
25:
26:  dbuf=AllocMem(512,MEMF_CHIP|MEMF_CLEAR);
27:  if(dbuf==NULL)
28:    {
29:    printf("Nicht genug Speicher!?\n");
30:    ende();
31:    }
32:  dpt=CreatePort("disk.port",NULL);
33:  drq=CreateStdIO(dpt);
34:
35:  if(strncmp(argv[1],"df",2)==NULL)
36:    { drive=(long)argv[1][2]-'0'; i=2; }
37:
38:  fehler=OpenDevice(TD_NAME,drive,drq,NULL);
39:
40:  if(fehler)
41:    {
42:    printf("Kann Geraet nicht ansprechen!\n");
43:    ende();
44:    }
45:
46:  if(i==2 && argc==2)
47:    {
48:    printf("Block-Nummer feht!\n");
49:    ende();
50:    }
51:
52:  drq->io_Length=1;
53:  drq->io_Command=TD_MOTOR;
54:  DoIO(drq);
55:
56:  for(kk=i;kk<argc;kk++)
57:    {
58:    blocknu=atoi(argv[kk]);
59:    if (1760<=blocknu || blocknu<=-1) blocknu=0;
60:    printf("Block %d:\n",blocknu);
61:
62:    drq->io_Command=CMD_READ;
63:    drq->io_Data=(APTR)dbuf;
64:    drq->io_Length=512;
65:    drq->io_Offset=blocknu*512;
66:    DoIO(drq);
67:
68:    for(n=0;n<=512;n+=16)
69:      {
70:      for(j=0;j<16;j++)
71:      printf("%02x ",dbuf[n+j]);
72:      printf("  ");
73:      for(j=0;j<16;j++)
74:        {
75:        if((dbuf[n+j]>0x1f && dbuf[n+j]<0x7f)
76:        ||  dbuf[n+j]>0x9f) printf("%c",dbuf[n+j]);
77:        else printf(".");
78:        }
79:      printf("\n");
80:      }
81:    }
82:   drq->io_Length=0;
83:   drq->io_Command=TD_MOTOR;
84:   DoIO(drq);
85:   ende();
86:  }
87: ende()
88:   {
89:   if(drq)
90:     {
91:     CloseDevice(drq);
92:     DeleteStdIO(drq);
93:     }
94:   if(dpt)  DeletePort(dpt);
95:   if(dbuf) FreeMem(dbuf,512);
96:   exit(NULL);
97:   }
```

beliebigen Tastendruck angehalten und mit <Return> oder <Backspace> fortgesetzt werden.

Nicht nur für den Bildschirm

Die Ausgabe der Blöcke läßt sich des weiteren auf andere logische Geräte umleiten. Beispielsweise kann der Block in ASCII-Form in einer Datei gesichert oder auf den Drucker ausgegeben werden.

```
BLOCK > prt: df0: 0
```

Der Block 0 der Diskette wird ausgedruckt.

```
BLOCK > ram:Bootblock df0: 0 1
```

Die Blöcke 0 und 1 werden in der Datei Bootblock im RAM in ASCII-Form gespeichert.

Auch für Festplatten

Das Programm BLOCK kann natürlich auch von Festplattenbesitzern genutzt werden. Jedoch müssen Sie einiges beachten. Als wichtigste Änderung muß in Zeile 40 *'TD_NAME'* in den jeweiligen Festplattentreiber umbenannt werden, beispielsweise *'(UBYTE*) "harddisk.divice"'*. Das *(UBYTE*)* sollte eingesetzt werden, um einem WARNING des Aztec-Compilers vorzubeugen. Zeile 37 und 38 können gelöscht oder als Kommentar deklariert werden, jedoch wäre eine entsprechende Anpassung besser. In Zeile 63 sollte die maximale Blockanzahl die der Festplatte angepaßte werden.

Das Programm ist mit dem Aztec C-Compiler erstellt. Folgender Compiler- und Linker-Aufruf sind zu ergreifen:

```
cc block.c +l
ln block.o -lc32
```

Eine Anpassung an den Lattice C Compiler dürfte aber keine Schwierigkeiten bereiten.

END of KICK

VON CHRISTIAN KELLER

WordPerfect im Detail

Teil 1
Typographie und Gestaltung

Das professionellste und sicher auch beste Textverarbeitungsprogramm für den AMIGA ist zur Zeit ohne Zweifel WordPerfect. Um Ihnen den Umgang damit zu erleichtern, wollen wir in dieser neuen Reihe auf einige grundlegende Betrachtungen der Benutzung eingehen.

Die Textgestaltung

Der Eindruck, den ein Brief hinterläßt, hängt, neben der korrekten Rechtschreibung, wesentlich von seiner Gestaltung ab. Texte, die an eine Seite des Blattes gedrückt sind, bis zum letzten Millimeter das Papier ausnutzen oder sonstwie windschief auf dem Papier erscheinen, werden sicherlich nicht den gewünschten Effekt beim Leser hervorrufen (Es sei denn, es handelt sich um eine Karnevalsrede).
Um die Ausgabe des Schreibens auf dem Drucker zu beeinflussen, stellt WordPerfect einige Hilfsmittel zur Verfügung.

Wysinwyg

In Bezug auf Formatierungen und manche Attribute ist WordPerfect nicht sehr 'aussagekräftig, was das normale Textfenster anbelangt. Am Bildschirm werden nur die Attribute fett, unterstrichen und kursiv angezeigt. Alle anderen Beeinflussungen des Textes können nur über die Funktion "Screen" in einem zweiten Fenster angezeigt werden. Ist man sich nicht im klaren. welche Funktionen gerade gesetzt sind, sollte man sich darüber vergewissern.

Papierformate

Das in Deutschland übliche Papier (DIN A4) hat eine Länge von 29.7 und eine Breite von 21 cm. Bei einer Zeichendichte von 6 cpi [Characters Per Inch (Inch = 25,4 mm)] und einem Zeilenabstand von 6 lpi (Lines Per Inch) haben also 70 Zeilen zu 83 Zeichen Platz auf dem Papier. Aussehen würde dies allerdings weniger gut, wie sich wohl jeder vorstellen kann. WordPerfect kann nun so eingestellt werden, daß automatisch bestimmte Ränder gelassen werden.
Erster Punkt der Justierung ist die Wahl des korrekten Papierformates. Diese geschieht mit Seitenformat-Seitenlänge. Benutzt man normales DIN A4-Papier, braucht man nur "Einzelblatt" anzuwählen.

WordPerfect stellt nun die Blattlänge auf 70 Zeilen (bei 6 lpi) und die Zeilenzahl auf 58 ein. Blattlänge bedeutet die "physikalische" Länge des Papiers, die Zeilenzahl gibt an, wieviele Zeilen auf einem Blatt gedruckt werden sollen. Diese Angaben werden immer auf 6 lpi bezogen und müssen nicht verändert werden, wenn man einen anderen Zeilenabstand einstellt. Im Zusammenhang mit der Seitenlänge stehen der obere und untere Blattrand. Explizit einstellbar ist nur der obere Blattrand. Der untere Rand ergibt sich aus der Blattlänge minus der Zeilenzahl und dem oberen Rand. Hat man mit Seitenformat-Oberer Blattrand diesen auf 12 (die Angabe erfolgt in 1/12 Zoll) eingestellt und das Einzelblattformat gewählt, erhält man einen unteren Rand von 1 Zoll oder 6 Zeilen (70 Zeilen - 58 Textzeilen - 6 Zeilen oberer Rand). WordPerfect geht beim Druck davon aus, daß das Papier bereits 1 Zoll eingezogen ist. Man sollte darauf achten, daß der automatische oder halbautomatische Einzelblatteinzug des Druckers dieses Maß einhält, da sonst die Formatierung des Textes nicht korrekt funktioniert.

Wählt man den oberen Rand größer, wird der gesamte Text auf dem Blatt nach unten verschoben, ohne daß die Zeilenzahl korrigiert wird. Wenn man mit dem Einzelblattformat also einen oberen Rand größer 24 einstellt, druckt WordPerfect über den unteren Rand des Papiers hinaus.

In der Horizontalen kann der Text natürlich auch beeinflußt werden. Mit Zeilenformat-Ränder kann man die Positionierung und die Breite des gedruckten Textes bestimmen. Dabei gibt es folgendes zu beachten: Die horizontalen Ränder beziehen sich nicht wie die vertikalen auf eine Grundgröße, sondern sind auf die aktuelle Zeichendichte bezogen. Druckt man eine Zeile in 10 CPI mit den Rändern 5 und 70, muß eine Zeile in 12 CPI mit den Rändern 6 und 84 gedruckt werden, so daß die linken und rechten Ränder beider Zeilen fluchten.

Beim Ausdruck gibt es noch eine Größe, die die Position des Textes auf dem Papier bestimmt. Das ist der Heftrand. Die Einstellung wird aber nicht in den Text übernommen, sondern nur beim aktuellen Druckjob verwendet. Texte auf Seiten mit ungeraden Seitennummern werden um den Heftrand nach links verschoben, Seiten mit gerader Nummer werden nach rechts bewegt. Dadurch ist es möglich, beidseitig bedruckte Blätter so zu gestalten, daß die Textblöcke auf beiden Seiten deckungsgleich übereinander liegen. Auch hier muß beachtet werden, daß die Zeilen nicht zu weit nach außen wandern. Der linke und der rechte Rand müssen mindestens so breit sein wie der Heftrand, da sonst über das Blatt hinausgeschrieben wird oder der Drucker von selbst in die nächste Zeile springt.

Drucker

Die oben beschriebenen Einstellungen ermöglichen es, den Text genau auf dem Blatt zu positionieren. Grundvoraussetzung dazu ist, daß der richtige Drucker installiert ist. Diesen Punkt sollte man peinlichst beachten, da sonst alle Formatierungen in WordPerfect nichts nützen. In der Druckerdatei wird man sicherlich den passenden Typ finden. Sollte dies nicht der Fall sein, kann man mit PrintDef einen vorhandenen Drukker an seine Hardwaregegebenheiten anpassen. Trotzdem gibt es noch einige Punkte zu beachten. Viele Drucker können zum Beispiel selbst linke und rechte Ränder setzen. WordPerfect nimmt dann natürlich den linken Rand des Druckers als seine Nullposition.

Schriftarten und Schrittweiten

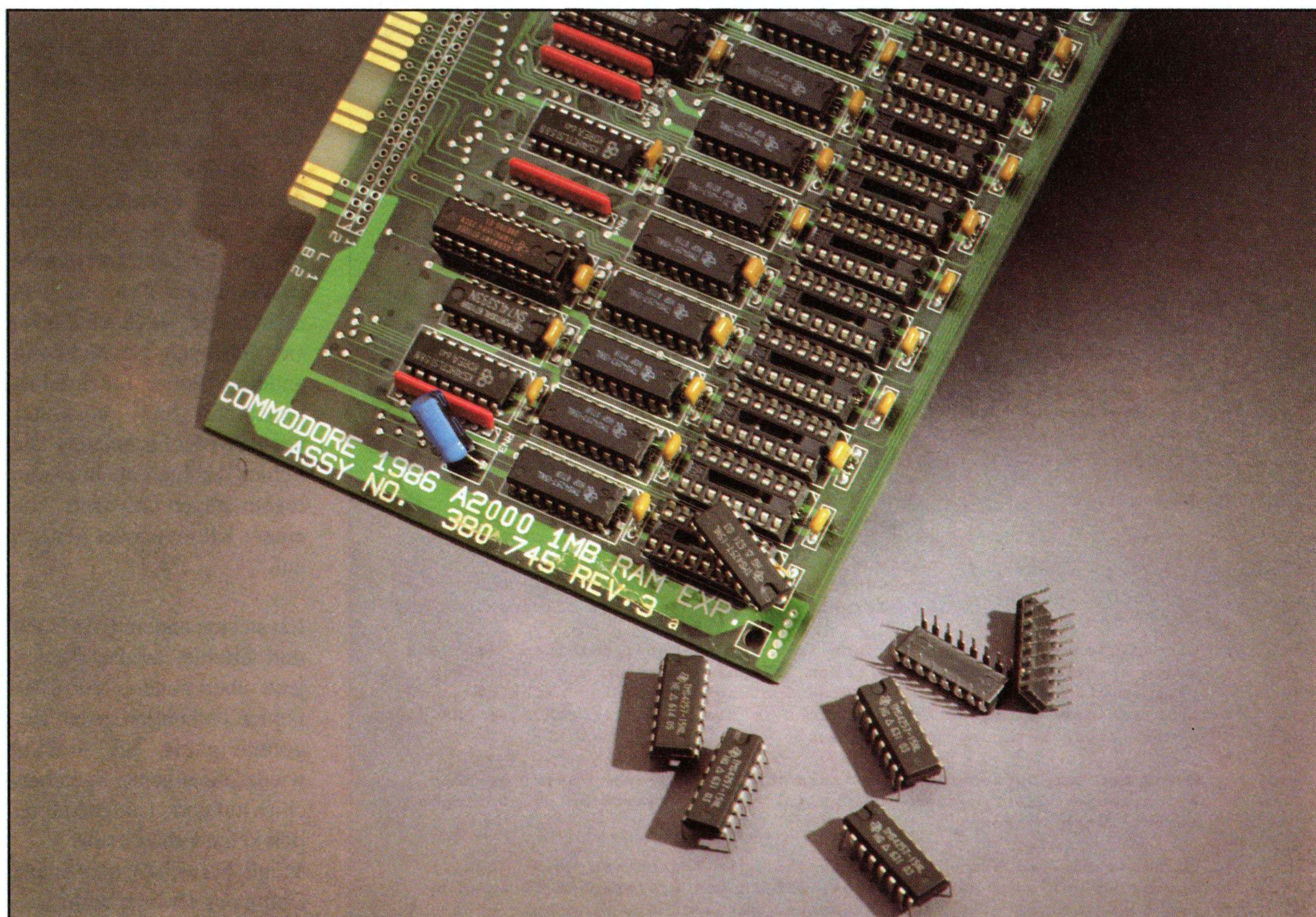
Die oben genannten Einstellungen sollten am Anfang des Textes aufgrund der verwendeten Zeichendichte (Pitch) gewählt werden. Innerhalb eines Textes ist es ratsam, mit Veränderungen des Pitch sparsam umzugehen, da dies der optischen "Qualität" des Textes zugute kommt (Ihr Kleingedrucktes dürfen Sie natürlich auch weiterhin mit 12 Zeilen und 50 Zeichen pro Inch ausdrucken). Ebenso knauserig sollte der Einsatz verschiedener Schriftarten erfolgen. Mit Fettdruck, Unterstreichungen und Kursivschrift stehen genügend Attribute zur Verfügung, um den Text zu gestalten. Bei der Erstellung von Büchern, Prospekten oder ähnlichem ist etwas mehr Freizügigkeit erlaubt als bei Briefen und dergleichen.

Innerhalb des "Textfensters" kann die Lage des Textes und seine Ausrichtung ebenfalls verändert werden. Am häufigsten wird man den Blocksatz einsetzen. Mit diesem Befehl werden alle Zeilen beim Druck so mit Lücken gespreizt, daß sie mit dem rechten Rand abschließen. Blocksatz wird nicht am Bildschirm angezeigt. Je nach Drucker werden Leerzeichen zwischen den Worten zusätzlich eingefügt oder, falls der Drucker dies unterstützt, kleinere Zwischenräume (sogenannte Micro-Spaces) zugesetzt. Die Micro-Spaces hinterlassen natürlich den besseren Eindruck, da man diese gleichmäßiger verteilen kann. In diesem Zusammenhang sei auf die Besonderheiten bei der Verwendung von Blocksatz mit Proportionalschrift hingewiesen. Diese Druckart erfordert die Verwendung eines speziellen PS-Fonts im Druckertreiber, der WordPerfect die verschiedenen Zeichenbreiten mitteilt. Nur so ist die korrekte Ausrichtung des Textes gewährleistet. Dieser Font befindet sich normalerweise unter Schrift 3. Den Pitch sollten sie auf 13 einstellen.
Man sollte Blocksatz nicht mit rechtsbündigem Drucken verwechseln. Dieses Kommando, das am Bildschirm angezeigt wird, läßt den linken Rand flattern. Rechtsbündig kann man zum Beispiel dazu verwenden, das Datum an den rechten Rand zu setzen.

Will man Überschriften optisch ansprechend plazieren, sollte man sie zentrieren. Wählt man diese Option, springt der Kursor in die Zeilenmitte, und der Text schiebt sich abwechselnd nach links und rechts. So ist eine gute optische Kontrolle möglich.

Damit wären die Gestaltungsmöglichkeiten beim Druck größtenteils angesprochen. Will man alle Optionen testen und seine Rechner-Drucker-Kombination bestmöglich ausnutzen, hat man sicher genug Beschäftigung bis zur nächsten Folge der WP-Praxis.

ENDE



VON MEINHARD ULLRICH

Amnesia machinae arithmeticae - Wenn Rechnern das Gedächtnis schwindet

Das geht Sie alle an!

Die Symptome

Glauben Sie nur nicht, Ihnen könnte so etwas nicht passieren. Es fängt ganz harmlos an: Sie spielen gerade Ballerschießbumm, brauchen nur noch 20000 Punkte für den High-Score und haben doch noch fünf Schiffe. Just in diesem Moment stürzt Ihr Rechner ab. Kismet, meinen Sie, und denken nicht länger darüber nach.

Eine Woche später laden Sie eine große Graphikanimationsdemo in Ihren Rechner. Sie haben ja eine Speichererweiterung auf 1.5 Megabyte, und das müßte laut Anleitung ausreichen. Geladen, gewartet, geduscht: Jetzt müßte das Programm eigentlich bereit sein. Doch Sie erleben plötzlich eine Graphikdemonstration ganz anderer Art. Rot, Blau, Grau, die LED blinkt, und der Rechner kommt nicht mehr hoch. Naja, gehe ich heute eben doch noch aus.

Als Sie dann aber zwei Tage später einen Artikel für die KICKSTART schreiben, ihn nach etwa fünf Stunden speichern wollen (was mir auch nur genau einmal in meinem Leben passiert - daß ich nicht zwischenspeichere, meine ich), und der Rechner ausgerechnet jetzt wieder seinen Urlaub einreicht, werden Sie langsam mißtrauisch.

Nein? Diese Situation ist Ihnen völlig fremd? Dann können Sie weiterblättern.

Die Diagnose

Vielleicht stimmt ja etwas mit meiner Speichererweiterung nicht. Aber nein, diese wird ja vom Hersteller ausgetestet und ist somit fehlerfrei. Außerdem hat sie ja auch eine Menge Geld gekostet. Und wenn doch, dann kann ich es ja auch nicht feststellen.

So, oder doch so ähnlich erging es mir vor kurzer Zeit. Aber da ich mir gerade einen C-Compiler gekauft hatte und

```c
/*******************************************************/
/*                                                     */
/*                      RAMTEST                        */
/*                                                     */
/* Geschrieben von Meinhard Ullrich                    */
/*                Hauptstraße 44                       */
/*                8771 Urspringen                      */
/*                                                     */
/* Copyright KICKSTART (MAXON Computer GmbH) 1988      */
/*                                                     */
/*******************************************************/


/*  Mit diesem Programm können Sie das Chipmem und das Fastmem Ihres
**  AMIGA testen.
**
**  Folgende Optionen werden unterstützt:
**    -f Fastmem, -c Chipmem;                  Standardeinstellung: -cf
**    -p Drucker, -b Bildschirm, -d Datei      Standardeinstellung: -b
**    -a ausführliche, -k kurze Informationen  Standardeinstellung: -k
**  Aufruf: RamTest {-{C|c|F|f|P|p|B|b|A|a|K|k}{[D|d] <Dateiname>}}
**
**  Beispiel: RamTest -fPa  gibt einen ausführlichen Test des
**                          Fastmems auf dem Drucker aus.
**
**            RamTest -dc df1:test.out
**                          testet das Chipmem und gibt Status- und
**                          Fehlermeldungen auf die Datei df1:test.out aus.
**
**  Sie sollten die Ausgabe nur dann in eine RAM-Datei umleiten, wenn Sie
**  genug an Chipmem und Fastmem haben. Hat das Programm das ganze Chip- oder
**  Fastmem alloziert, und vom anderen Speicherbereich ist nicht genug da,
**  dann schlagen einige Schreibversuche in die RAM-Datei fehl, so daß diese
**  danach etwas verkrüppelt sein kann.
**
**  Wurde bei den Optionen sowohl "Ausführlich" als auch "Kurz" gewählt,
**  so verwendet das Programm die Option, die als letzte in der Optionen-
**  zeichenkette vorkommt.
**
**  Übrigens sollten Sie - sofern Sie sich die Ergebnisse am Bildschirm
**  anschauen wollen - darauf achten, daß in den Preferences 80 Zeichen
**  pro Zeile eingestellt sind. Ansonsten bekommen Sie eine sehr häßliche
**  Bildschirmdarstellung, aber was red' ich...
**
**  Zur Funktion:
**  Das Programm beschreibt nacheinander den gesamten freien Speicher mit
**  den Testmustern 0000, 1010, 0101 und 1111 und überprüft danach, ob der
**  Speicher noch diese Muster enthält.
**
**  Verwendeter Compiler:
**  Das Programm ist für den Aztec-Compiler geschrieben.
**
**  Ich schlage vor, die Objektdatei mit der Aztec-Linker-Option +C
**  zu binden, weil damit Programm und Daten zur Laufzeit ins Chipmem
**  geladen werden, welchem ich mehr traue als dem Fastmem (vor allem
**  den RAM-Erweiterungen). Besteht besonderer Anlaß, dem Chipmem zu miß-
**  trauen, so sollten Sie eine Version mit der +F(astmem)-Option erstellen.
*/


#include "libraries/dosextens.h"
#include "exec/types.h"
#include "exec/memory.h"
#include "clib/macros.h"
#include "functions.h"


#define        MAX_SEGMENTE 100
#define        BIN 1
#define        DEZ 2
#define        HEX 3

typedef struct
{
    ULONG Erster_Fehler;
    ULONG Letzter_Fehler;
    ULONG Fehlerzaehler;
} Fehlerrapport;

typedef struct                          /* Bitfeld */
    {
       unsigned Chipmem         : 1;
       unsigned Fastmem         : 1;
       unsigned Drucker         : 1;
```

möglichst schnell C lernen wollte, habe ich dieses Programm zusammengehackt.

Und wie ich schon bald merkte, stimmte wirklich etwas mit meiner Speichererweiterung nicht. Viele Speicherzellen auf der Speichererweiterung konnten 1'en und 0'en nicht halten. Vielleicht ließen sie sich auch gar nicht mit anderen Werten beschreiben (stuck-at-0 bzw. stuck-at-1 nennt das der Fachmann). Die Fehler tauchten oft in einem regelmäßigen Abstand von einem Kilobyte oder ähnlich auf.

Ich steckte andere RAMs auf und erlebte wieder Fehler, jetzt aber anderer Art. Die Fehler entbehrten jeder Regelmäßigkeit. Ich steckte wieder die eigenen Speicherchips auf und - ich fand plötzlich keine Fehler mehr. Möglicherweise waren meine Chips nur etwas krumm aufgesteckt? Ich steckte die anderen wieder ein, sehr sorgfältig diesmal, und konnte die alten Fehler genau reproduzieren.

Bei meinen eigenen RAM's hingegen konnte ich es anscheinend steuern, ob Fehler auftraten. Diese waren also möglicherweise in Ordnung, waren nur unsauber in den Fassungen. (Übrigens handelt es sich um 256er RAMs!)

Rezept und Medizin

Wechseln Sie also, so Sie Ihre Speichererweiterung betreffend einen Verdacht hegen, die Chips 3 x täglich vor dem Essen aus. Oder aber benutzen Sie das unten abgedruckte Programm zur Kontrolle einer Vitalfunktion Ihres AMIGA.

Leider kann das Programm

```
         unsigned Bildschirm  : 1;
         unsigned Datei       : 1;
         unsigned Ausfuehrlich: 1;
       } Optionstyp;

/***********************************************/
/***              Globale Variablen          ***/
/***********************************************/

BOOL                                 Geschwaetzig;
char                                 Textpuffer[80];

extern    struct FileHandle   *Open();
          struct FileHandle   *Bildschirm, *Drucker,
                                           *Datei;

struct             /* belegt 8*MAX_SEGMENTE Byte */
{
    ULONG * Tab_Speicheranfang;
    ULONG Tab_Speicherlaenge;
}                                Tabelle[MAX_SEGMENTE];

LONG                             Gesamtfehler = 0;

Optionstyp                       Option = {0,0,0,0,0,0};

ULONG                            Testzahl[4] =
{
0x00000000, /* 00000000000000000000000000000000 */
0xaaaaaaaa, /* 10101010101010101010101010101010 */
0x55555555, /* 01010101010101010101010101010101 */
0xffffffff  /* 11111111111111111111111111111111 */
};



/***********************************************/
/***         Prozeduren und Funktionen       ***/
/***********************************************/



       /****************************/
VOID  Speicherfreigabe(Segmente)
       /****************************/

 /************************************************
 Eingabe:        (unsigned long) Segmente
 Funktionen:  FreeMem  [Amiga-Standardfunktion]
 Effekt:
 Die Funktion gibt belegte Speichersegmente frei,
 deren Adressen und Längen in der globalen
 Variablen Tabelle gespeichert sind. <Segmente>
 bezeichnet Anzahl der freizugebenden Segmente.
 ************************************************/
    ULONG  Segmente;

{
    /* Global: Tabelle[] */
    SHORT i;

    for (i = 1; i <= Segmente; i++)
        FreeMem(Tabelle[i-1].Tab_Speicheranfang,
                Tabelle[i-1].Tab_Speicherlaenge);
}



       /****************************/
void  SchreibeMedien(Zeichenkette)
       /****************************/

 /************************************************
 Eingabe:        (char *) Zeichenkette
 Funktionen:  Write           [Amiga-Standardfunktion]
              strlen               [Aztec-Bibliothek]
              Speicherfreigabe [interne Funktion]
   Effekt:
   Die Prozedur gibt die übergebene <Zeichenkette>
   auf alle geöffneten Kanäle aus, die da wären:
   Bildschirm, Drucker oder eine Datei.
 ************************************************/

    STRPTR  Zeichenkette;

{
    /* Global: Bildschirm, Drucker, Datei */
    STATIC BOOL  MissErfolg;

    if (Bildschirm)
       Write(Bildschirm, Zeichenkette,
                      (LONG)strlen(Zeichenkette));

    if (Drucker &&!MissErfolg)
    {
       MissErfolg =
          (Write(Drucker, Zeichenkette,
                 (LONG)strlen(Zeichenkette)) == -1);

       if (MissErfolg &&!Bildschirm)/* Sag's uns */
          printf("Ab hier also ohne Drucker...\n");
    }

    if (Datei)
       Write(Datei, Zeichenkette,
                      (LONG)strlen(Zeichenkette));
}



        /***********************/
STRPTR  Konvertiere(Zahl, Modus)
        /***********************/

 /*********************************************
 Eingabe:        (unsigned long) Zahl,
                 (int)           Modus
 Ausgabe:        (char *)        Textpuffer
 Funktionen:     sprintf  [Aztec-Bibliothek]
                 strcpy   [Aztec-Bibliothek]
 Effekt:
 Die Funktion erhält eine <Zahl> und wandelt sie
 dem <Modus> entsprechend in eine Zeichenkette
 vom Typ HEX, BIN oder DEZ um.
 ********************************************/
    ULONG  Zahl;
    int    Modus;

{
    /* Global: Textpuffer[] */
    SHORT i;

    if       (Modus == HEX)
       sprintf(Textpuffer, "%lx",Zahl);
    else if (Modus == DEZ)
       sprintf(Textpuffer, "%ld", Zahl);
    else if (Modus == BIN)
    {
       /* Bilde Maske mit einer 1 an der ent-
          sprechenden Stelle, vergleiche         **
       ** mit der auszuwertenden Zahl und setze
          im Textpuffer an der entsprechenden    **
       **  Stelle eine '1' bzw. eine '0'.        */
       for (i=0; i<32; i++)
          Textpuffer[i] = (((0x80000000 >> i) &
                         Zahl) == 0)? '0': '1';
       Textpuffer[32] = '\0';
    }
    else
       strcpy(Textpuffer, "!! Falscher Modus!!!");

    return (Textpuffer);
    /* Rückgabe eines Funktionswertes nur aus     **
    ** ästhetischen Gründen; eigentlich ist die   **
    ** Adresse der globalen Variablen Textpuffer **
    ** auch global bekannt (vgl. Aufr. der Fkt.).*/
}



       /******************************************/
VOID  Teste_Speicherblock(Speicheranfang,
                   Speicherlaenge, Sammelmeldung)
       /******************************************/

 /*********************************************
 Eingabe:        (unsigned long *) Speicheranfang,
                 (unsigned long)   Speicherlaenge
```

```
Ausgabe:      (unsigned long)     ErsterFehler,
                                  LetzterFehler,
                                  Fehlerzaehler
Funktionen:  SchreibeMedien     [interne Funktion]
             Konvertiere        [interne Funktion]
             MIN, MAX           [Makros, def. in clib/macros.h]

Effekt: Die Funktion bekommt mit <Speicheranfang> einen Zeiger auf einen
        Speicherblock und in <Speicherlaenge> die Länge des Blocks.
        Sie füllt den Speicherblock jeweils ganz mit einem der vier
        Testmuster und prüft ihn danach wieder durch.
        Sie gibt dabei Statusmeldungen auf alle geöffnete Kanäle aus.
        Die Ausgabeparameter werden in eine Struktur vom Typ Fehlerrapport
        geschrieben, deren Adresse in <Sammelmeldung> übergeben wurde.
****************************************************************/

  ULONG        *  Speicheranfang;
  ULONG           Speicherlaenge;
  Fehlerrapport * Sammelmeldung;

{
  /* Global: Geschwaetzig, Testzahl[] */
  UCOUNT j;
  LONG lokaler_Fehler;
  REGISTER LONG Testmuster;
  REGISTER ULONG * Speicherzeiger;
  REGISTER ULONG * Speicherende;
  REGISTER ULONG Erster_Fehler  = 0xffffffff;
  REGISTER ULONG Letzter_Fehler = 0x00000000;
  REGISTER ULONG Fehlerzaehler  = 0L;

  Speicherende = Speicheranfang + (Speicherlaenge/4);
     /* Ein Langwort = 4 Bytes */

  for (j = 0; j <= 3; j++)       /* 4 Testmuster durchprobieren */
  {
     Testmuster = Testzahl[j];
     if(Geschwaetzig)
     {
        SchreibeMedien("  Beschreibe den Speicher mit  %");
        SchreibeMedien(Konvertiere(Testmuster, BIN));
        SchreibeMedien("\n");
     }

     /* Ein Dank an Claus Brod, der durch einen wertvollen Tip */
     /* zur Optimierung dieser Funktion beigetragen hat:       */
     /* Speicherzeiger statt Speicherarithmetik!               */

     /* Wir kleistern zuerst den ganzen reservierten Speicher voll ... */
     Speicherzeiger = Speicheranfang;
     while (Speicherzeiger < Speicherende)
     {
        *Speicherzeiger = Testmuster;   /* etwas günstiger als:          */
        Speicherzeiger++;               /* *Speicherzeiger++ = Testmuster */
     }

     /* ... Und lesen ihn erst dann wieder aus */
     Speicherzeiger = Speicheranfang;
     lokaler_Fehler = 0L;
     while (Speicherzeiger < Speicherende)
     {
        if (*Speicherzeiger!= Testmuster)
        {
           if(Geschwaetzig)
           {
              SchreibeMedien("     Adresse $");
              SchreibeMedien(Konvertiere((ULONG)Speicherzeiger, HEX));
              SchreibeMedien(", gelesen  %");
              SchreibeMedien(Konvertiere(*Speicherzeiger, BIN));
              SchreibeMedien("\n");
              lokaler_Fehler++;
           }

           Erster_Fehler  = MIN(Erster_Fehler,  (ULONG)Speicherzeiger);
           Letzter_Fehler = MAX(Letzter_Fehler, (ULONG)Speicherzeiger);
           Fehlerzaehler++;
        }
        Speicherzeiger++;
     }
     if (Geschwaetzig && (lokaler_Fehler!= 0))
        SchreibeMedien("\n");
  }

  /* Warum nicht gleich mit folgender Struktur arbeiten? **
```

den Speicher Ihres Rechners noch nicht heilen, (Auf Update warten!). Mit seiner Hilfe kann aber eventuell schleichender Gedächtnisschwund frühzeitig erkannt und der Rechner einer weiteren Behandlung zugeführt werden.

Zusammensetzung

Außer ein paar unvermeidlichen Emulgatoren, Geschmacksverstärkern und naturunidentischen Aromastoffen enthält das Programm nur einen Wirkstoff: die Prozedur *Teste_Speicherblock()*. Diese Prozedur bekommt von *Speichertest()* den Anfang eines Speicherblocks und dessen Länge. Nacheinander beschreibt sie diesen Block mit den Bitmustern 0000, 1010, 0101 und 1111 und vergleicht hernach den Speicher wieder mit den Mustern. Falls Fehler auftreten, gibt die Prozedur eine Fehlermeldung auf alle mittels Optionen gewählten Geräte (Drucker, Bildschirm, Datei) aus. Am Ende schreibt sie in eine Struktur, deren Adresse ihr beim Aufruf übergeben wurde, wieviele Fehler in diesem Segment auftauchten, und grenzt den fehlerhaften Speicher ein.

Das Präparat enthält kein 7-Chlor-1,3-dihydro-3-hydroxy-5-phenyl-1,4- benzodiazepin-2-on und kein Kakaopulver.

Gegenanzeigen, Neben- und Wechselwirkungen

Bei bestimmten Formen von Charakterschwäche, wie zum Beispiel dem Compilieren mit einem raubkopierten C-Compiler, können in selten Fällen massive Anfälle von Gewissensbissen auftreten, die erfahrungsgemäß bei den meisten

```
    ** 1. ist das Arbeiten mit Registervariablen schneller,**
    ** 2. ist es so etwas übersichtlicher:                */
    Sammelmeldung->Erster_Fehler  = Erster_Fehler;
    Sammelmeldung->Letzter_Fehler = Letzter_Fehler;
    Sammelmeldung->Fehlerzaehler  = Fehlerzaehler;

}


      /******************/
VOID  Schliesse_Medien()
      /******************/

 /*****************************************************************
 Funktionen:  Close  [Amiga-Standardfunktion]

 Effekt: Alle offenen Kanäle werden wieder geschlossen.
 *****************************************************************/

{
    /* Global: Bildschirm, Drucker, Datei */

    if (Bildschirm)
       Close(Bildschirm);
    if (Drucker)
       Close(Drucker);
    if (Datei)
       Close(Datei);
}


      /*********************************/
VOID  Gebrauchsanweisung(Programmname)
      /*********************************/

 /*****************************************************************
 Eingabe:      (char *) Programmname
 Funktionen:  printf

 Effekt: Gibt Gebrauchsanweisung für das Programm ins aktuelle
         CLI-Fenster aus. Es wird dabei berücksichtigt, ob der
         Benutzer das Programm umgetauft hat (<Programmname>).
 *****************************************************************/

    STRPTR  Programmname;

{
    printf("\n\n");
    printf("Schreibweise: %s ", Programmname);
    printf("{-{C|c|F|f|P|p|B|b|A|a|K|k}{[D|d] <Dateiname>}}\n\n");

    printf("Die Optionen haben folgende Bedeutung:\n");

    printf("  C,c:  Das Chipmem wird getestet.\n");
    printf("  F,f:  Das Fastmem wird getestet.\n");
    /* Standard: Chipmem und Fastmem */

    printf("  P,p:  Die Ausgabe erfolgt auf den Drucker.\n");
    printf("  B,b:  Die Ausgabe erfolgt auf dem Bildschirm.\n");
    printf("  D,d:  Die Ausgabe erfolgt in eine Datei, die erfragt wird.\n");
    /* Standard: Bildschirmausgabe  */

    printf("  A,a:  Ausführliche Fehlermeldungen.\n");
    printf("  K,k:  Kurzer Fehlerbericht.\n");
    /* Standard: Kurzausführung                              **
    ** Falls beide Optionen gewählt wurden, kommt das zum Tragen, **
    ** was zuletzt spezifiziert wurde.                       */
}




      /****************************************/
VOID  Setze_Optionen(Spezifikation, Vorgabe)
      /****************************************/

 /*****************************************************************
 Eingabe:      (char *) Spezifikation,
               (BOOL)   Vorgabe
 Funktionen:  strlen  [Aztec-Bibliothek]

 Effekt: Ist <Vorgabe> wahr, wird das globale Bitfeld Optionen
         entsprechend der Optionenzeichenkette <Spezifikation>
         - andernfalls mit den Standardoptionen - besetzt.
```

Anwendern in kurzer Zeit wieder abklingen.

Das Programm sollte nie gleichzeitig mit anderen wichtigen Anwendungen laufen. Es ist nicht restlos ausgetestet - auf Tierversuche habe ich gänzlich verzichtet - und ich kann somit nicht für es garantieren. Im übrigen ist es sowieso besser, wenn möglichst wenig Speicher belegt ist, damit sehr viel zum Test zur Verfügung steht Ich schlage also vor, es direkt nach dem Booten zu starten. Am besten jedoch ist es, von einer fast nackten Diskette zu booten, auf der nur die Startup-Sequence mit dem Aufruf *Ram-Test [Optionen]* sowie das Programm selbst und eventuell ein Druckertreiber steht.

Bei meinen exzessiven Testläufen habe ich indes keine Fehler mehr feststellen können. Falls jemand andere Erfahrungen macht, wäre ich aber für eine Nachricht sehr dankbar.

Dosierungs-anleitung

Soweit vom Discdoctor nicht anders verordnet, sollten Sie dieses Programm bei abnormalem Verhalten einmal, bei Zweifeln oder einem hartnäkkigen Verdacht auch mehrmals laufen lassen. Sie haben dabei die Wahl zwischen verschiedenen Optionen. So können Sie mit (*B*)ildschirm, (*D*)atei und (*P*)rinter ein Ausgabemedium oder auch mehrere selektieren. Haben Sie unter anderem *D* angegeben, so müssen Sie nach den Optionen noch eine Datei mit Pfad und Namen angeben, in die Status- und Fehlermeldungen abgespeichert werden sollen.
Hier gilt es jedoch, eine Besonderheit zu beachten: Sie sollten nämlich nur von der RAM-

```
*************************************************/

    STRPTR  Spezifikation;
    BOOL    Vorgabe;

  {
    /* Global: Option, Geschwaetzig */
    int i;

    if (Vorgabe)
       for (i = 1; i < strlen(Spezifikation); i++)
          switch(*(Spezifikation + i))
          {
             case 'c':
             case 'C':
                Option.Chipmem = 1;
                break;

             case 'f':
             case 'F':
                Option.Fastmem = 1;
                break;

             case 'p':
             case 'P':
                Option.Drucker = 1;
                break;

             case 'b':
             case 'B':
                Option.Bildschirm  = 1;
                break;

             case 'd':
             case 'D':
                Option.Datei = 1;
                break;

             case 'a':
             case 'A':
                Option.Ausfuehrlich = 1;
                break;

             case 'k':
             case 'K':
                Option.Ausfuehrlich = 0;
                break;

             default:
                printf("Option %c nicht vorgesehen!
                  \n", *(Spezifikation + i));
                exit(FALSE);
                break;
          }

    if ((Option.Datei == 0) && (Option.Bildschirm
            == 0) && (Option.Drucker == 0))
       Option.Bildschirm = 1;/* Standardeinstell.*/

    if((Option.Chipmem == 0)&&(Option.Fastmem == 0))
       Option.Chipmem = Option.Fastmem = 1;
                      /* Standardeinstellung */
    Geschwaetzig = (BOOL)Option.Ausfuehrlich;
  }

     /************************/
  VOID  Oeffne_Medien(Dateiname)
     /************************/

 /*************************************************
 Funktionen: printf,
             Open          [Amiga-Standardfunktion],
             Exit          [Amiga-Standardfunktion],
             Schliesse_Medien   [interne Funktion]
 Effekt:
 Entsprechend den gewählten Optionen bzw. der
 Standardeinstellung werden die Kanäle Datei,
 Drucker und Bildschirm für die Ausgabe geöffnet.
 Soll auch eine Datei geöffnet werden, verwendet
 die Funktion den Namen, der in der Kommandozeile
 *************************************************/
    STRPTR  Dateiname;

  {
   /* Global: Option, Bildschirm, Drucker, Datei */

    if (Option.Datei == 1)
    {
       Datei = Open(Dateiname, MODE_NEWFILE);
       if (!Datei)
       {
          printf("Leider kann ich diese Datei nicht
                                         öffnen.\n");
          exit(FALSE);
       }
    }

    if (Option.Bildschirm == 1)
    {
       Bildschirm = Open("RAW:1/1/639/199/RamTest",
                                       MODE_NEWFILE);
       if (!Bildschirm)
       {
          printf("CLI-Fenster läßt sich nicht
                  öffnen. Tut mir leid...\n");
          Schliesse_Medien();
          exit(FALSE);
       }

    }

    if (Option.Drucker == 1)
    {
       Drucker = Open("PRT:", MODE_OLDFILE);
       if ((Write(Drucker, "\0", 1L)) == -1L)
       {
          printf("Sind Sie sicher, daß Sie
                 überhaupt einen Drucker haben?\n");
          Schliesse_Medien();
          exit(FALSE);
       }
    }
  }

     /**************/
  VOID  Speichertest()
     /**************/

 /*************************************************
 Funktionen:SchreibeMedien       [interne Funktion],
            AvailMem       [Amiga-Standardfunktion],
            AllocMem       [Amiga-Standardfunktion],
            Konvertiere          [interne Funktion],
            Teste_Speicherblock[interne Funktion],
            Schliesse_Medien     [interne Funktion],
            printf,
            exit           [Amiga-Standardfunktion],
            Speicherfreigabe     [interne Funktion]
 Effekt:
  In dieser Prozedur wird die meiste Arbeit getan:
  Es wird das Chipmem und/oder Fastmem getestet.
  Dabei werden Status- und Fehlermeldungen
  auf die offenen Kanäle ausgegeben.
 *************************************************/
  {
    STRPTR  Speicherbereich;
    LONG  Fehlerzaehler, Speicherart, Laenge;
    LONG  getesteter_Speicher, Belegte_Segmente;
    LONG  * Speicheranfang;
    Fehlerrapport  Sammelmeldung;

    SchreibeMedien("   ******************* \n");
    SchreibeMedien("   *** Speichertest *** \n");
    SchreibeMedien("   ******************* \n\n");

    while ((Option.Chipmem == 1) ||
                             (Option.Fastmem == 1))
    {
       if (Option.Chipmem == 1)
       {
          Speicherbereich = "Chipmem";
          Speicherart = MEMF_CHIP | MEMF_PUBLIC;
          Option.Chipmem = 0;
       }
       else
       {
          Speicherbereich = "Fastmem";
          Speicherart = MEMF_FAST | MEMF_PUBLIC;
```

```
                Option.Fastmem = 0;
            }

        SchreibeMedien("=======================\n");
        SchreibeMedien(" Ich teste jetzt das ");
        SchreibeMedien(Speicherbereich);
        SchreibeMedien("\n");
        SchreibeMedien("=====================\n\n");

        Belegte_Segmente = Fehlerzaehler = getesteter_Speicher = 0L;

        while ((Laenge = AvailMem(Speicherart | MEMF_LARGEST))!= NULL)
            if ((Speicheranfang = AllocMem(Laenge, Speicherart))!= NULL)
            {
                if (Belegte_Segmente < MAX_SEGMENTE)
                {
                    Tabelle[Belegte_Segmente].Tab_Speicheranfang = Speicheranfang;
                    Tabelle[Belegte_Segmente].Tab_Speicherlaenge = Laenge;
                }
                Belegte_Segmente++;

                SchreibeMedien("\nTeste den Speicher von $");
                SchreibeMedien(Konvertiere((ULONG)Speicheranfang, HEX));
                SchreibeMedien(" bis $");
                SchreibeMedien(Konvertiere((ULONG)(Speicheranfang + Laenge/4),
                                                              HEX));
                SchreibeMedien(" (Länge: ");
                SchreibeMedien(Konvertiere((ULONG)Laenge, DEZ));
                SchreibeMedien(" Bytes)\n");

                Teste_Speicherblock(Speicheranfang, Laenge, &Sammelmeldung);
                /* Das letzte Argument bezeichnet ein Feld, **
                ** indem ich danach die Ergebnisse erwarte. */

                Fehlerzaehler += Sammelmeldung.Fehlerzaehler;
                getesteter_Speicher += Laenge;

                SchreibeMedien("  In diesem Segment befanden sich ");
                SchreibeMedien(Konvertiere(Sammelmeldung.Fehlerzaehler, DEZ));
                SchreibeMedien(" Fehler.\n");

                if (Sammelmeldung.Fehlerzaehler!= 0L)
                {
                    SchreibeMedien("  Untere Fehlergrenze:   $");
                    SchreibeMedien(Konvertiere(Sammelmeldung.Erster_Fehler, HEX));
                    SchreibeMedien("\n");
                    SchreibeMedien("  Obere Fehlergrenze:    $");
                    SchreibeMedien(Konvertiere(Sammelmeldung.Letzter_Fehler, HEX));
                    SchreibeMedien("\n");
                    SchreibeMedien("  Länge dieses Segments: ");
                    SchreibeMedien(Konvertiere(Sammelmeldung.Letzter_Fehler -
                                       Sammelmeldung.Erster_Fehler, DEZ));
                    SchreibeMedien(" Bytes\n");
                }

            }
            else
            {
                Schliesse_Medien();
                printf("System gibt mir freien Speicher nicht!\n");
                exit(FALSE);
            }

        SchreibeMedien("\n\n Im ");
        SchreibeMedien(Speicherbereich);
        SchreibeMedien(" traten insgesamt (bei vier Testmustern) ");
        SchreibeMedien(Konvertiere(Fehlerzaehler, DEZ));
        SchreibeMedien(" Fehler auf.\n");

        SchreibeMedien(" Es wurden insgesamt ");
        SchreibeMedien(Konvertiere(getesteter_Speicher, DEZ));
        SchreibeMedien(" ($");
        SchreibeMedien(Konvertiere(getesteter_Speicher, HEX));
        SchreibeMedien(") Bytes getestet.\n");

        SchreibeMedien(" Der Speicher war auf ");
        SchreibeMedien(Konvertiere(Belegte_Segmente, DEZ));
        SchreibeMedien(" Segmente verteilt.\n\n\n\n");

        Gesamtfehler += Fehlerzaehler;

        Speicherfreigabe(MIN(Belegte_Segmente, MAX_SEGMENTE));

    }
```

Disk Gebrauch machen, wenn Sie noch viel Chipmem und Fastmem zur Verfügung haben. Eine offene RAM-Datei alloziert bei Schreibzugriffen dynamisch Speicherplatz. Wenn der RAM-Test aber gerade die letzten freien Segmente des Chipmems oder des Fastmems prüft, ist von dem jeweiligen Speicher nichts mehr übrig, und die RAM-Datei verliert den Streit um den letzten Teil des Speichers.

Wenn jetzt vom jeweils anderen Speicher (Fastmem oder Chipmem) noch genügend große Segmente vorhanden sind, sucht sich die Datei dort Speicherplatz, wenn nicht, schlagen ein paar Schreibversuche fehl, die RAM-Datei wird verkrüppelt.

Wenn das Programm beide Speicherbereiche untersuchen soll, prüft es erst einen Bereich durch und gibt diesen Speicher wieder frei, bevor es sich um den anderen kümmert.

Als weitere Optionen stehen *F* für Fastmem und *C* für Chipmem zur Verfügung, außerdem steht *A* für ausführliche Informationen, während *K* eine geringere Information zur Folge hat.

Zum Beispiel bringt "*RamTest -cfab*" einen ausführlichen Testbericht von Fastmem und Chipmem auf den Bildschirm. "*RamTest -d df1:Ramfehler*" gibt den Speichertest in die spezifizierte Datei aus. Für den Fall, daß einige Optionen nicht angegeben werden, werden Standardeinstellungen benutzt: Wurde kein Ausgabemedium festgelegt, geht die Ausgabe an den Bildschirm, entfiel die Spezifizierung des Speicherbereichs, wird Chipmem und Fastmem getestet. Außerdem ist die Option: kurzgehaltene Statusmeldungen voreingestellt.

```
}

	/****************/
VOID main(argc, argv)
	/****************/

/***************************************************************
 Funktionen: index                  [Aztec-Bibliothek],
             Gebrauchsanweisung [interne Funktion],
             Setze_Optionen     [interne Funktion],
             Oeffne_Medien      [interne Funktion],
             Speichertest       [interne Funktion],
             Read               [Amiga-Standardfunktion],
             Schliesse_Medien   [interne Funktion]

 Effekt: "Und hurre hurre, hopp, hopp, hopp,
          Gings fort in sausendem Galopp,
          Daß Roß und Reiter schnoben
          Und Kies und Funken stoben..."
         [aus: Lenore, Gottfried August Bürger]
 ***************************************************************/

   int     argc;
   STRPTR  argv[];

 {
   STRPTR Spezifikation, Dateiname;
   char  Puffer[2];      /* Es gibt Tasten, die zwei Werte liefern */

   /* Falls Optionen spezifiziert wurden, muß genau dann ein Dateiname */
   /* angegeben worden sein, wenn die Option d (D) verwendet wurde.    */
   if (
         (argc!= 1)
         &&!((*argv[1] == '-')
            &&( ((argc == 2) && ( (index(argv[1], 'd') == 0L)
                                  &&(index(argv[1], 'D') == 0L)))
               ||((argc == 3) && ( (index(argv[1], 'd')!= 0L)
                                  ||(index(argv[1], 'D')!= 0L)))
               )
            )
      )
   {
      Gebrauchsanweisung(argv[0]);
      exit(FALSE);
   }

   Spezifikation = (argc >= 2)? argv[1]: "dummy";
   Setze_Optionen(Spezifikation, (argc >= 2));

   Dateiname = (argc == 3)? argv[2]: "noname";
   Oeffne_Medien(Dateiname);

   Speichertest();

   if (Bildschirm)          /* Warte auf Tastendruck */
      Read(Bildschirm, Puffer, 1L);

   Schliesse_Medien();

   if (Gesamtfehler!= 0L)
      printf ("\nSpeichertest fand Fehler! \n");
 }
```

Zur Beachtung

Das Programm ist für den Aztec-Compiler geschrieben. Aus mangelnder Kenntnis anderer Compiler habe ich nicht auf größtmögliche Portabilität geachtet.
Trotzdem glaube ich nicht, daß ich sehr spezifische Funktionen dieses Compilers verwendet habe.

Wenn Sie wie ich vor allem Ihr Fastmem testen wollen, sollten Sie die Objektdatei unbedingt mit der Linkeroption *+C* binden, weil dann Programm und Daten zur Laufzeit ins Chipmem geladen werden, so daß erstens mehr Speicher im Fastmem zum Testen freibleibt, und zweitens das Programm selbst ordentlich laufen kann.

Das Programm ist für Kinder zugänglich aufzubewahren!

ENDE

VON ROLF BECK

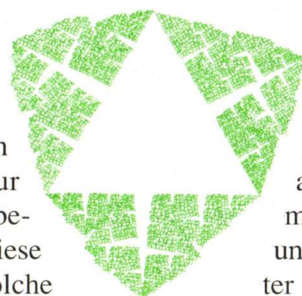
GEOMETRIE ZAUBER

Peano- und Kochkurven

Daß sich aus einfachen geometrischen Figuren fantastische fraktale Figuren 'zaubern' lassen, beweist das abgedruckte Programm. Die große Flexibilität und Komplexität des Programms bringt immer neue mathematische Figuren auf den AMIGA-Bildschirm. Lassen Sie sich verzaubern, kehren Sie ein in die Welt der einfachen Geometrie und erzeugen Sie selbst Koch- oder Peanokurven.



Das Programm erzeugt fraktale Kurven, die aus einem Initiator und der wiederholten Anwendung eines Generators entstehen. Das Prinzip ist folgendes: Eine Gerade (der Initiator), die von zwei Punkten begrenzt wird - nennen wir sie A und B - wird durch einen Linienzug (Generator genannt) ersetzt, dessen Anfangs- und Endpunkte wieder A und B sind. Durch diesen Generator steht nun anstatt einer einzigen Gerade eine ganze Anzahl davon zur Verfügung, von denen jede wieder von 2 Punkten begrenzt ist. Diese Prozedur kann man mit der ganzen Figur beliebig oft wiederholen, wobei diese immer komplexer wird. Solche



Figuren werden Peano- oder Kochkurven genannt, je nachdem, ob sie ebenfüllend sind oder nicht. Dies ist aber ein rein mathematisches Problem und soll hier nicht weiter interessieren.



Die Grundform der Kurve hängt von dem Initiator ab, der nicht unbedingt nur eine Gerade sein muß; einfache geometrische Gebilde wie Dreieck, Quadrat etc. liefern viel interessantere Ergebnisse. Weitere Einzelheiten über dieses Thema findet man in "Die fraktale Geometrie der Natur" von B. Mandelbrot.

Bedienung des Programms

Der Aufruf erfolgt vom CLI aus durch:

Koch <Initiator> <Generator> <Level> <Modus>
Dabei besitzen die Parameter folgende Bedeutung:

Initiator: eine ganze Zahl von 0 bis 7. Sie gibt die Grundform an.

0 großes Dreieck
1 Rechteck
2 lange Linie
3 zwei sich überdeckende lange Linien
4 Sechseck
5 kleines Dreieck
6 Quadrat aus acht Linien (2 pro Seite)
7 zwei kurze Linien

Generator: eine Zahl von 0 bis 15

Die Form der Generatoren kann mit "*Koch 2 <Gen> 1 0*" angezeigt werden; eine Beschreibung mit Worten wäre zu umständlich.

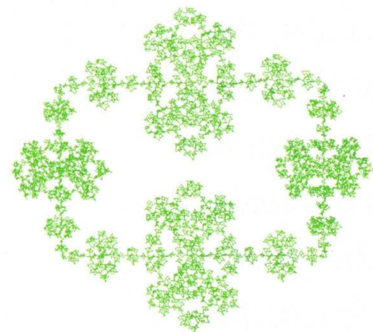
Level: Eine Zahl von 1 bis 8, die angibt, wie oft der Prozeß der Generatoranwendung wiederholt werden soll. Je höher sie gewählt wird, desto länger benötigt natürlich die Berechnung.

Modus: eine Zahl von 0 bis 3. Ein Generator kann vor seiner Anwendung auch gespiegelt werden, dadurch ändert sich das Aussehen der Kurve mehr oder weniger stark.

0 *normal* (keine Spiegelung)
1 *gespiegelt* (wird während der ganzen Berechnung beibehalten)
2 *wechselnd* (vor jeder Iteration wird zwischen normal und gespiegelt umgeschaltet)
3 *Zufall* (vor jeder Generatoranwendung wird durch Zufall entschieden, ob gespiegelt wird). Modus 3 bringt bei weitem die interessantesten Ergebnisse hervor. Die Kurve ähnelt dann Inseln oder Küstenlinien.

Beendet wird das Programm, indem man das Koch-Fenster anklickt und die ESC-Taste drückt.
Wenn Sie die untenstehenden Parameter verwender erhalten Sie besonders schöne Bilder:

0 0 5 3   2 2 7 0
5 5 6 0   4 15 5 0
4 5 6 1   6 10 4 3

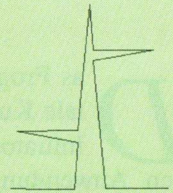


```
/*
 *  KOCH-KURVEN
 *  Autor: Rolf Beck
 * zeichnet Koch-, Peano- und andere Fraktale
   Kurven
 * Compilieren: cc +p Koch (Aztec C V3.6a)
 * ln Koch -lm132 -lc132
 *
 * Aufruf: Koch <Generator> <Initiator>
               <Iterationen> <Modus>
 *        Modus: Art des Generators:
 *               0=normal,    1=gespiegelt,
 *               2=wechselnd, 3=zufällig
 *        Beenden mit ESC
 * (c) MAXON Computer GmbH
```

```
 * KICKSTART 1989
 */

 #include <math.h>
 #include <stdio.h>
 #include <exec/types.h>
 #include <exec/memory.h>
 #include <functions.h>
 #include <intuition/intuition.h>

 #define SCREEN_WIDTH 640
 #define SCREEN_HEIGHT 530
 #define PSIZE (ULONG)sizeof(struct Point)
 #define PI 3.1415927
 #define L1 2.64575
```



```
#define UP 0            /* Generator-Modi */
#define DOWN 1
#define CHANGE 2
#define RANDOM 3

#define Abstand(x1,y1,x2,y2)
    sqrt((double)((x1)-(x2))*((x1)-(x2))+((y1)-
                        (y2))*((y1)-(y2)))

struct GfxBase *GfxBase;
struct IntuitionBase *IntuitionBa

struct Window *Wind;
struct Screen *Scre;
struct RastPort *Rp;
struct ViewPort *Vp;

struct NewWindow NW = {
    0,0,SCREEN_WIDTH,SCREEN_HEIGHT,-1,-1,
    VANILLAKEY,
    BACKDROP|BORDERLESS|SMART_REFRESH,
    NULL,NULL,NULL,0,NULL,0,0,SCREEN_WIDTH,
    SCREEN_HEIGHT,CUSTOMSCREEN };

struct NewScreen NS = {
    0,0,SCREEN_WIDTH,SCREEN_HEIGHT,2,0,1,
    HIRES|LACE,CUSTOMSCREEN,NULL,
    (UBYTE *)"Koch-Kurven    1988 von Rolf Beck "
    ,NULL,NULL };

struct Point     /* Punktkoordinaten */
{
    SHORT x;
    SHORT y;
};

UWORD Mult; /* Anzahl Elemente des Generators */
UWORD Gen;  /* Nummer des Generators          */
UWORD In;   /* Nummer des Initiators          */
UBYTE Modus=UP; /* Art der Generatoranwendung */
BYTE MM=1;      /* Wert der Generatoranwendung */
struct Point *A1,*A2;  /* dynamische Arrays   */

struct Point Initiator [][9] =
/* Format: {{X1,Y1},{X2,Y2}},{nächster
    Generator},...                            */
{
   {{320,20},{600,410},{40,410},{320,20}},
        /* Dreieck groß */
   {{160,160},{480,160},{480,370},{160,370},
    {160,160}},
        /* Rechteck */
   {{0,265},{639,265}},      /* Linie waagrecht */
   {{0,265},{639,265},{0,265}},/*doppelte Linie*/
   {{220,100},{420,100},{540,265},
   {420,430},{220,430},{100,265},{220,100}},
        /* Sechseck */
   {{320,100},{470,360},{170,360},{320,100}},
        /* Dreieck klein */
    {{140,80},{320,80},{500,80},{500,260},
     {500,440},
       {320,440},{140,440},{140,260},{140,80}},
        /* Quadrat unterteilt */
    {{0,265},{320,265},{639,265}},
        /* Linie unterteilt */
};
UBYTE InitAnz[]= /* Anzahl der Punkte jedes
                    Initiators                */
{
    4,5,2,3,7,4,9,3
};

float Generator [][36] =
/* Format: { Länge (Nenner eines Bruches),
                   Winkel in Bogenmaß, ... },... */
{
    { 3,0 ,3,-PI/3 ,3,PI/3 ,3,0 },
    { 4,0, 4,-PI/2, 4,0, 4,PI/2, 4,PI/2, 4,0, 4,-
      PI/2, 4,0 },
    { 4,0, 3,-17*PI/36, 3,17*PI/36 },
    { 6,0, 6,-PI/2, 6,-PI/2, 6,0, 6,0, 6,PI/2,
      6,PI, 6,PI/2, 6,0, 6,0, 6,PI/2, 6,PI, 6,PI/
      2, 6,0,
      6,0, 6,-PI/2, 6,-PI/2, 6,0 },
    { 1.73205,-PI/6, 1.73205,PI/6 },
    { 12/5,0, 5/2,-17*PI/36, 5/2,17*PI/36 },
    { 3.4641,-PI/6, 3.4641,PI/6, 3.4641,PI/6,
      3.4641,-PI/6 },
    { 5,0, 5,PI/2, 5,0, 5,-PI/3, 5,PI/3,
      5,0, 5,PI/2, 5,0 },
    { 2,PI/2, 2,0, 2,-PI/2, 2,-PI/3, 2,PI/3 },
    { 5,0, 5,PI/2, 5,PI*5/6, 5,PI/6,
           5,PI/2, 5,0, 5,PI/3, 5,-PI/3, 5,0,
           5,-PI/2, 5,-PI/6, 5,-PI*5/6, 5,-PI/2,
           5,0 },
    { 4,PI/3, 4,0, 4,-PI/3, 4,-PI/3, 4,0,
      4,PI/3 },
    { 3,0, 3,PI/3, 3,-PI/3, 3,-PI, 3,-PI/3,
       3,PI/3, 3,0 },
    { L1,-PI/9, L1,PI/4, L1,8*PI/9, L1,5*PI/9,
      L1,-PI/9,
      L1,-PI/9, L1,-4*PI/9 },
    { 3,PI/3, 3,PI/3, 3,0, 3,-PI/3, 1.5,-5*PI/6,
      3,0, 3,0 },
    { 3,0, 6,PI/2, 6,PI, 6,PI/2, 3,0, 6,-PI/2,
                6,PI, 6,-PI/2, 3,0,
      6,-PI/2, 6,PI, 6,-PI/2, 3,0, 6,PI/2, 6,PI,
      6,PI/2, 3,0 },
    { 2.12132,PI/4, 3,-PI/2, 2.598076,-PI/6,
      2.598076,PI/6 }
};
UBYTE GenAnz [] = /* Mult-Wert für jeden
                     Generator */
{
    4,8,4,18,2,4,4,8,5,14,6,7,7,7,17,4
};

void Cleanup()
{
    if(Wind) CloseWindow(Wind);
    if(Scre) CloseScreen(Scre);

    if(GfxBase) CloseLibrary(GfxBase);
    if(IntuitionBase)CloseLibrary(IntuitionBase);
}

void OpenAll ()
{
    if(!(IntuitionBase=(struct IntuitionBase *)
        OpenLibrary("intuition.library",0L)))
        Cleanup();
    if(!(GfxBase=(struct GfxBase *)
        OpenLibrary("graphics.library",0L)))
        Cleanup();

    if(!(Scre=OpenScreen(&NS)))    Cleanup();
    NW.Screen=Scre;
    if(!(Wind=OpenWindow(&NW)))    Cleanup();

    Rp=Wind->RPort;
    Vp=&(Scre->ViewPort);

    SetRGB4(Vp,0L,0L,0L,0L);SetRGB4(Vp,1L,2L,
                                      11L,4L);
    SetRGB4(Vp,2L,8L,8L,8L);SetRGB4(Vp,3L,12L,
                                      6L,5L);
}
```

```
float Winkel(x1,y1,x2,y2)
UWORD x1,y1,x2,y2;
/* berechnet Winkel einer Geraden mit der
   Horizontalen                                */
{
   SHORT x,y;
   float Betrag,winkel;

   x=x2-x1; y=y2-y1;
   if(!(Betrag=sqrt((double)(x*x+y*y))))
      return(0);
   winkel=acos((double)(x/Betrag));
   if (y1>y2)
      return(2*PI-winkel);
   else
      return(winkel);
}

void Generate(ind)
UCOUNT ind;
/* holt 2 Punkte aus A1,
   verwendet Generator, legt Punkte in A2 ab  */
{
   struct Point P;        /*aktueller Punkt   */
   register float L,W,l,w;
/*Ausgangswinkel,-länge ,aktueller Winkel,Länge */
   register UCOUNT i1,i2; /*Schleifenvariable */

   P=A1[ind];
   L=Abstand (P.x,P.y,A1[ind+1].x,A1[ind+1].y);
   W=Winkel  (P.x,P.y,A1[ind+1].x,A1[ind+1].y);

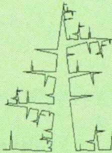
   for(i2=0,i1=1+ind*Mult;i2<2*Mult-2;i2+=
                                   2,i1++)
   {
      l=L/Generator [Gen][i2];
      w=W+Generator [Gen][i2+1]*MM;
      P.x= P.x+l*cos((double)w);
      P.y= P.y+l*sin((double)w);
      A2[i1]=P;
   }
   A2[i1]= A1[ind+1];
}

void main (argc,argv)
int argc;
char *argv[];
{
   UCOUNT n;     /* Anzahl der Iterationen     */
   ULONG Anzahl;/* Anzahl Punkte im Initiator */
   register UCOUNT i;  /* Schleifenzähler      */
   struct IntuiMessage* msg;
   ULONG class;           /* Bestandteile der   */
   USHORT code;           /* IDCMP-Message      */
   BOOL Continue=TRUE;    /* kennzeichnet
                             Programmende       */

   if(argc!=5 || (In=atoi(argv[1]))>7 ||
   (Gen=atoi(argv[2]))>15 ||(Modus=atoi(argv
                             [4]))>RANDOM)
   {
      puts("Usage: Koch <Initiator> <Generator>
                        <Level> <Modus>\n");
      exit(10);
   }
   n=atoi(argv[3]);

   OpenAll();
   srand((unsigned int)time((long *)0L));

   A1=A2=0L;
```



```


   Mult=GenAnz[Gen]; Anzahl=InitAnz[In];
   if(!(A1=
      AllocMem((long)Anzahl*PSIZE,(ULONG)
      MEMF_FAST|MEMF_PUBLIC)))
      Cleanup();
   movmem((char *)&Initiator[In][0],
              (char *)A1,(int)(Anzahl*PSIZE));

   switch(Modus)
   {
    case UP:   MM=1;
               break;
    case DOWN: MM=-1;
   }
   while (n–)
   {
      if(!(A2=AllocMem((long)(Mult*(Anzahl
            1)+1)*PSIZE,
      (long)MEMF_FAST|MEMF_PUBLIC)))
         Cleanup();
      A2[0]= A1[0];

      if (Modus==CHANGE)
         MM=-MM;

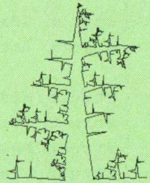
      for(i=0;i<Anzahl-1;i++)
      {
         if (Modus==RANDOM)
            MM=(rand()&1)? 1:-1;
         Generate(i);
      }

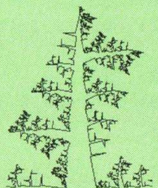
      FreeMem(A1,(long)Anzahl*PSIZE);
      A1= A2;
      Anzahl= Mult*(Anzahl-1)+1;

      SetRast(Rp,0L);
      SetAPen(Rp,1L);
      Move(Rp,(long)A1[0].x,(long)A1[0].y);
      PolyDraw(Rp,(long)Anzahl,A1);
   }
   FreeMem(A1,(long)Anzahl*PSIZE);

   while(Continue)
   {
      Wait(1L<<Wind->UserPort->mp_SigBit);
      while(msg=(struct IntuiMessage*)GetMsg
                     (Wind->UserPort))
      {
         class=msg->Class;
         code=msg->Code;
         ReplyMsg(msg);

         switch(class)
         {
          case VANILLAKEY:
            if (code==27)  /* ESC */
               Continue=FALSE;
            break;
          default:
            fputs("Unbekannte Message\n",
                                    stderr);
         }
      }
   }
   Cleanup();
} /* LISTING ENDE */
```





VON REINER MOLL

KEINE ANGST VOR VIREN

Grundlegendes über Computerviren

Lästige kleine Plagegeister haben sich breitgemacht. Von manchen Computerbesitzern sicher noch gar nicht bemerkt, haben sie sich in vielen Diskettensammlungen eingenistet - die Computerviren. Was zunächst als ein Scherz einer Crackergruppe begann, hat sich inzwischen zu einer Plage ausgeweitet, die viele Computerfreaks nervt, verunsichert und ängstigt. Wissen Sie doch, daß Viren einen großen Schaden anrichten können, was Zeitungsmeldungen immer wieder belegen. Legte doch ein Virus vor nicht allzulanger Zeit den Computer des Pentagon lahm.

Der erste AMIGA-Virus

Bei uns hier in Deutschland war der erste bekanntgewordene AMIGA-Virus der von der Crackergruppe SCA. Dabei handelte es sicch um ein kleines Programm, das von Zeit zu Zeit einen harmlosen Lauftext auf dem Bildschirm ausgab. Nach jeweils genau 16 Bootvorgängen (Resets) lief dieser Text über den Bildschirm und zeigte damit das Vorhandensein des Virus an.
Es wurde kein weiteres Unheil angerichtet, außer daß der Lade-Bootblock überschrieben wurde. Bei den nächsten Resets lief zunächst wieder alles normal ab. Das Neue und Raffinierte war, daß sich das Programm selbsttätig von Diskette zu Diskette weiterverbreiten konnte, ohne daß es der gutmütige AMIGA-Anwender bemerkte. Die Vorgehensweise des Virus war folgendermaßen:

Die Fortpflanzung

Wurde ein Warmstart ("Booten") mit einer "infizierten" Diskette durchgeführt, nistete sich der Virus sofort im Speicher des Computers ein und legte sich in Lauerstellung. Wenn man nun irgendwann später mit einer anderen Diskette bootete, z.B. um ein anderes Computerspiel zu spielen, schrieb sich bei diesem Bootvorgang der Virus aus

```
Der SCA-Virus

0000: 444F5300 37FCBB02 43485721 41FAFFF2   DOS.7...CHW!A...
.
.
.
02E0: 01800800 8605FFFE 01800D51 8A05FFFE   ..........Q....
02F0: 01800000 FFFFFFFE 002BFFFA FFD06772   .........+....gr
0300: 61706869 63732E6C 69627261 72790064   aphics.library.d
0310: 6F732E6C 69627261 72790000 20536F6D   os.library.. Som
0320: 65746869 6E672077 6F6E6465 7266756C   ething wonderful
0330: 20686173 20686170 70656E65 64D2AA2D    has happened..-
0340: 17596F75 7220414D 49474120 69732061   .Your AMIGA is a
0350: 6C697665 20212121 BEA04113 616E642C   live!!!..A.and,
0360: 20657665 6E206265 74746572 2E2E2E50    even better...P
0370: 50041F53 6F6D6520 6F662079 6F757220   P..Some of your
0380: 6469736B 73206172 6520696E 66656374   disks are infect
0390: 65646E32 5A0E6279 20612056 49525553   edn2Z.by a VIRUS
03A0: 20212121 8C783216 416E6F74 68657220   !!!.x2.Another
03B0: 6D617374 65727069 65636520 6F668232   masterpiece of.2
03C0: 32165468 65204D65 67612D4D 69676874   2.The Mega-Might
03D0: 79205343 41202121 DC6E0000 4EF90000   y SCA!!.n..N...
03E0: 00004121 53434121 53434121 53434121   ..A!SCA!SCA!SCA!
03F0: 53434121 53434121 53434121 53434121   SCA!SCA!SCA!SCA!
```

Bild 1: Der SCA-Virus war der erste Virus, der auf dem AMIGA sein Unwesen getrieben hat. Er entpuppte sich als realtiv harmlos.

```
Der BYTE BANDIT-Virus


0000: 444F5300 41D25F27 00000370 6000003E     DOS.A._'...p'..>
0010: 56697275 73206279 20427974 65202020     Virus by Byte
0020: 42616E64 69742069 6E202039 2E38372E     Bandit in  9.87.
0030: 4E756D62 6572206F 66202020 20202020     Number of
0040: 636F7079 73203A00 0000020C 48E77F7F     copys:.....H...
0050: 2C780004 33FC4000 00DFF09A 062E0001     ,x..3.@........
0060: 0126302E 0094C0BC 000000F0 B07C00F0     .&0.........|..
0070: 66000040 41EE0142 20502268 00182A49     f..@A..B P"h..*I
0080: 41FAFF7E 45FA0348 12D8B5C8 66FA41FA     A..~E..H....f.A.
0090: FF702008 41FA000A 22089280 DBC14ED5     .p .A..."....N.
00A0: 41FA032C 428020C0 20C06100 003E6100     A..,B. . .a..>a.
00B0: 00AC43FA 030D4EAE FFA04A80 67000024     ..C...N...J.g..$
.
.
.
```

Bild 2: Der BYTE BANDIT-Virus schaltete nach sechs Minuten die Bildschirm-DMA ab. Die Programmierer waren aber so gütig und haben eine Hintertür offengelassen.

dem Speicher unbemerkt auf die zuletzt eingelegte Diskette. Voraussetzung:

1. Der Computer wurde nicht abgeschaltet, nachdem sich der Virus im Speicher eingenistet hatte, und

2. die Diskette, mit der zuletzt gebootet wurde, die also "infiziert" wurde, war nicht schreibgeschützt. Nun sieht man ja nicht ohne weiteres eine Veranlassung, wenn man ein anderes Spiel spielen möchte, den Computer zwischendurch auszuschalten. Außerdem erfordern es viele Programme, den Schreibschutz der Diskette zu deaktivieren, z.B. weil man sich nur so in die auf der Diskette gespeicherte Highscore-Liste eines Spieles eintragen kann. Das hat zu einer sehr weiten Verbreitung des SCA-Virus' geführt.

Im Lauf der Zeit...

Aber, wie gesagt, der SCA-Virus war noch ganz 'harmlos'. Inzwischen gibt es Viren, die erheblich unangenehmer sind: BYTE-BANDIT schaltet nach sechs Minuten den Bildschirm dunkel (Erkennungsmerkmal!). Der ahnungslose Anwender dachte zunächst, daß sich der AMIGA wohl ins Reich der Träume zurückgezogen hätte und nur durch einen Reset wieder erweckt werden konnte. Diese eigenartige Erscheinung trat aber immer häufiger auf. "Vielleicht ist die Hardware defekt", dachten sich viele AMIGA-Anwender mit Schrecken. Doch der freundliche AMIGA-Händler konnte auch nach noch so langem Suchen keinen Defekt feststellen. Nur durch Zufall konnte die merkwürdige Krankheit analysiert werden. Der BYTE BANDIT-Virus war das Übel aller Dinge. Die Programmierer des BYTE BANDIT-Virus' haben aber gütigerweise eine kleine Hintertür offen gelassen. Durch Drücken der Tastenkombinationen in der Reihenfolge ALT, AMIGA-links, SPACE, AMIGA-rechts und ALT wird der Bildschirm wieder für weitere sechs Minuten normal. Doch schlimmere Viren ließen nicht lange auf sich warten.
Aber zunächst zu zwei Abarten.

LSD und WARHAWK...

... sind reine Mutationen des SCA-Virus'. Es erfolgte lediglich eine Änderung des SCA-Textes, so daß eine neue Meldung ausgegeben wurde. Beim WARHAWK-Virus wurde anstatt eines einfachen Textes (nach 16 Resets) ein kleiner Effekt präsentiert. Zu beachten ist bei diesen Viren, daß sie von manchen Virus-Protektoren nicht erkannt werden. Erhöhte Vorsicht ist geboten.

DASA

Der DASA-Virus erzeugt nach nicht vorhersehbarer Zeit einen READ/WRITE-Error auf der Diskette. Die betroffene Diskette ist erst nach Neuformatierung wieder verwendbar, d.h. alle darauf befindlichen Daten sind verloren. Erkennungsmerkmal: Bei einer plötzlichen READ/WRITE-Meldung des Systems kann der DASA-Virus in Aktion gewesen sein. Gegenmaßnahmen, auf die ich noch zu sprechen komme, sollten sicherheitshalber sofort vorgenommen werden.
Wenn man Glück im Unglück hatte, konnte der Anwender einige wichtige Daten retten. Nicht so beim...

...BYTE WARRIOR

Der BYTE WARRIOR gehört zu den schlimmsten AMIGA-Viren, die z.Z. existieren. Im Gegensatz zu den bisher kennengelernten macht er nicht einmal vor den bisher unbeachteten Festplatten halt. Der BYTE WARRIOR formatiert wichtige Teile des Speichermediums. Spätestens wenn der AMIGA-Anwender sich mit diesem Virus infiziert hat, wird er merken, daß man mit Computerviren nicht spaßen sollte. Die Verbreitung findet ebenfalls über den Bootblock statt und unterscheidet sich von diesem Gesichtspunkt aus betrachtet nur wenig von den bisher bekannten Viren.

```
Der DASA-Virus


0000: 444F5300 64116DEC 09D5A859 48E7FFFE     DOS.d.m....YH...
0010: 6100002E 4CDF7FFF 43FA001A 4EAEFFA0     a...L...C...N...
0020: 4A806700 000C2040 20680016 70004E75     J.g... @ h..p.Nu
0030: 70FF4E75 646F732E 6C696272 61727900     p.Nudos.library.
0040: 207C0007 F80043FA FFB8303C 00FF20D9      |....C...0<.. .
0050: 51C8FFFC 207C0007 FA66227C 0007FC00     Q... |...f"|....
0060: 303C0096 12180A01 00FFE619 12C151C8     0<..........Q.
0070: FFF44AAE 002A6600 000A4AAE 002E6700     ..J..*f...J...g.
0080: 00084EB9 0007F8A8 2C790000 00042D7C     ..N.....,y....-|
0090: 0007F94C 02264EAE FD9C2D40 022A2D7C     ...L.&N...-@.*-|
00A0: 0007F96E FE3A4E75 42AE002A 42AE002E     ...n.:NuB..*B...
00B0: 303C0017 41EE0022 4241D258 51C8FFFC     0<..A.."BA.XQ...
00C0: 44415341 3081323C 000813FC 000200BF     DASA0.2<.......
00D0: E001303C 300051C8 FFFE13FC 000000BF     ..0<0.Q........
00E0: E001303C 300051C8 FFFE51C9 FFDE23FC     ..0<0.Q...Q...#.
00F0: 0007FA62 00DFF0D0 33FC0002 00DFF0D4     ...b....3......
```

Bild 3: Der DASA-Virus gehört schon zu den schlimmeren AMIGA-Viren. Plötzlich autretende READ/WRITE-Errors können dem DASA-Virus zu verdanken sein.

```
Der NORTH STAR-Virusprotector

0000: 444F5300 37FCBB02 5E4142F8 601A6000   DOS.7...^AB.'.'.
0010: 00784E6F 72746820 53746172 0002000A   .xNorth Star....
0020: 00030004 00000002 0CB94E6F 72740007   ..........Nort..
0030: EC126618 0CB95374 61720007 EC18660C   ..f...Star....f.
0040: 0C790002 0007EC1C 6D026022 41FAFFB2   .y......m.'"A...
.
.
.
0170: 4CDF3FFF 4E750CAC 00BFE001 0048660A   L.?.Nu.......Hf.
0180: 06790001 0007EC20 60740CAC 42616E64   .y..... `t..Band
0190: 0020660A 06790001 0007EC22 60600CAC   . f..y....."''..
01A0: 00BFE001 00A4660A 06790001 0007EC24   ......f..y.....$
01B0: 604C0CAC 6F746563 0024660A 06790001   `L..otec.$f..y..
01C0: 0007EC26 605A4E75 0CAC4E6F 72740010   ...&'ZNu..Nort..
01D0: 66140CAC 53746172 0016660A 0C6C0002   f...Star..f..l..
01E0: 001A6702 6D024E75 6100FF76 0CB90000   ..g.m.Nua..v....
01F0: 00000007 EFE8671C 61000124 60EA6100   ......g.a..$'.a.
.
.
.
0350: 6C696272 61727900 00C03CB4 00C00276   library...<....v
0360: 00500C56 49525553 20446574 65637465   .P.VIRUS Detecte
0370: 64206F6E 20446973 6B212053 54415246   d on Disk! STARF
0380: 4952452F 4E4F5254 48205354 41520000   IRE/NORTH STAR..
0390: 007E0C4F 4C442041 6E746956 69727573   .~.OLD AntiVirus
03A0: 2E202053 54415246 4952452F 4E4F5254   .  STARFIRE/NORT
03B0: 48205354 41520000 00500C4D 7920416E   H STAR...P.My An
03C0: 74695669 72757320 69732042 65747465   tiVirus is Bette
03D0: 72212053 54415246 4952452F 4E4F5254   r! STARFIRE/NORT
03E0: 48205354 41520000 0000001C 646F732E   H STAR......dos.
03F0: 6C696272 61727900 4EF90000 00000000   library.N.......
```

Bild 4: Der NORTH STAR ANTI-Virus wurde geschrieben, um das epidemieartige Verbreiten von Viren einzudämmen.

Die Virus-Schützer

Der NORTH STAR ANTIVIRUS Überwacht den Boot-Sector auf das Vorhandensein des SCA- und des BYTE BANDIT-Virus' und meldet ihn gegebenenfalls mit einem Warnfenster, einem sogenannten ALERT oder er überschreibt den Bootblock einschließlich des Fremdvirus' mit sich selbst - falls die betroffene Diskette nicht schreibgeschützt ist. Dadurch macht er die Diskette unter Umständen unbrauchbar. Erkennungsmerkmal: Flakkern der Ladelampe bei Betätigung des Joystick-Feuerknopfes. Ein weiterer Virus-Protector, der mittlerweile in verschiedenen Versionen die AMIGA-Szene bevölkert, nennt sich Z-System. Die aktuelle ist z.Z. die Version 4.0. Er schreibt sich ebenfalls auf den Bootblock und zeigt das Vorhandensein eines Virus' an.

Der erste Schutz!

Die bisher bekanntgewordenen Amiga-Viren arbeiten alle ähnlich wie der SCA-Virus. Weil diese Arbeitsweise bekannt ist, ist auch auf einfache Art ein wirksamer Schutz möglich: man muß jede Diskette, die man neu in seine Sammlung aufnimmt, sofort untersuchen, ob sie womöglich infiziert ist. Ist der Befund positiv, muß der Bootblock der Diskette vom Virus befreit werden. Dazu ist folgende Vorgehensweise erforderlich:

Man startet den AMIGA mit der Original-Workbench, von der man weiß, daß sie garantiert virusfrei ist (Eine Kopie, von der man dies ebenfalls 100%ig weiß, tut es auch - aber das weiß man nur 100%ig, wenn man sie nach dem Kopiervorgang niemals ohne Schreibschutz benutzt hat.). Unter Benutzung dieser Workbench formatiert man eine neue Diskette (entweder CLI-Befehl "*format*" oder Icon der neuen Diskette auf der Intuition-Oberfläche einmal anklicken und Pulldown-Menüauswahl '*Disk/Initialize*' anklicken). Diese neu formatierte Diskette ist jetzt vom AMIGA lesbar, aber sie kann so nicht als Bootdiskette benutzt werden. Nun geht es weiter:

- die zu überprüfende Diskette einlegen und damit booten (Reset).
  Ist die Diskette "virusinfiziert", wird dabei der Virus aus dem Bootblock in den Computerspeicher kopiert.

- Diskette austauschen gegen die zu überprüfende Diskette

```
Der Z-SYSTEM-Virusportector


0000: 444F5300 50564C2E 00000370 60000022   DOS.PVL....p'.."
0010: 20535953 54454D20 5A205649 52555320    SYSTEM Z VIRUS
0020: 2050524F 54454354 4F522056 342E3020    PROTECTOR V4.0
0030: 41FAFFCE 43F90007 F400303C 010022D8   A...C.....0<..".
0040: 51C8FFFC 2C790000 000442AE 002E2D7C   Q...,y....B...-|
.
.
.
0280: 0003001C 237C0000 04000024 237C0007   ....#|.....$#|..
0290: F4000028 42A9002C 4EAEFE38 224D337C   ...(B..,N..8"M3|
02A0: 0004001C 4EAEFE38 6000FF5A 005A1057   ....N..8'..Z.Z.W
02B0: 61726E69 6E673A20 54686973 20646973   arning: This dis
02C0: 6B206973 20696E66 65637465 64207769   k is infected wi
02D0: 74682061 20566972 75732120 00010012   th a Virus! ....
02E0: 204C6566 74204D6F 75736562 7574746F    Left Mousebutto
02F0: 6E3A204B 696C6C20 74686520 56697275   n: Kill the Viru
0300: 732C2052 69676874 204D6F75 73656275   s, Right Mousebu
0310: 74746F6E 3A20436F 6E74696E 75650000   tton: Continue..
0320: 00141E54 68697320 6469736B 20697320   ...This disk is
0330: 696E6665 63746564 20776974 68207468   infected with th
0340: 65202741 4E544927 56697275 73206F66   e 'ANTI'Virus of
0350: 204E6F72 74685374 61720000 006E1E54    NorthStar...n.T
0360: 68697320 6469736B 20636F6E 7461696E   his disk contain
0370: 7320616E 206F6C64 20566972 75735072   s an old VirusPr
0380: 6F746563 746F722E 000001AE 4EAEFD9C   otector.....N...
0390: 00964EAE FD9C0056 4BF900DF 80246F74   ..N....VK....$ot
03A0: 65638046 4EAEFD9C 802C332E 30202020   ec.FN....,3.0
03B0: 20202020 20202020 20202020 20202020
.
.
.
```

Bild 5: Der Z-SYSTEM-Virus ist ebenfalls ein Virusprotector und ist der z.Z. 'leistungsfähigste'.

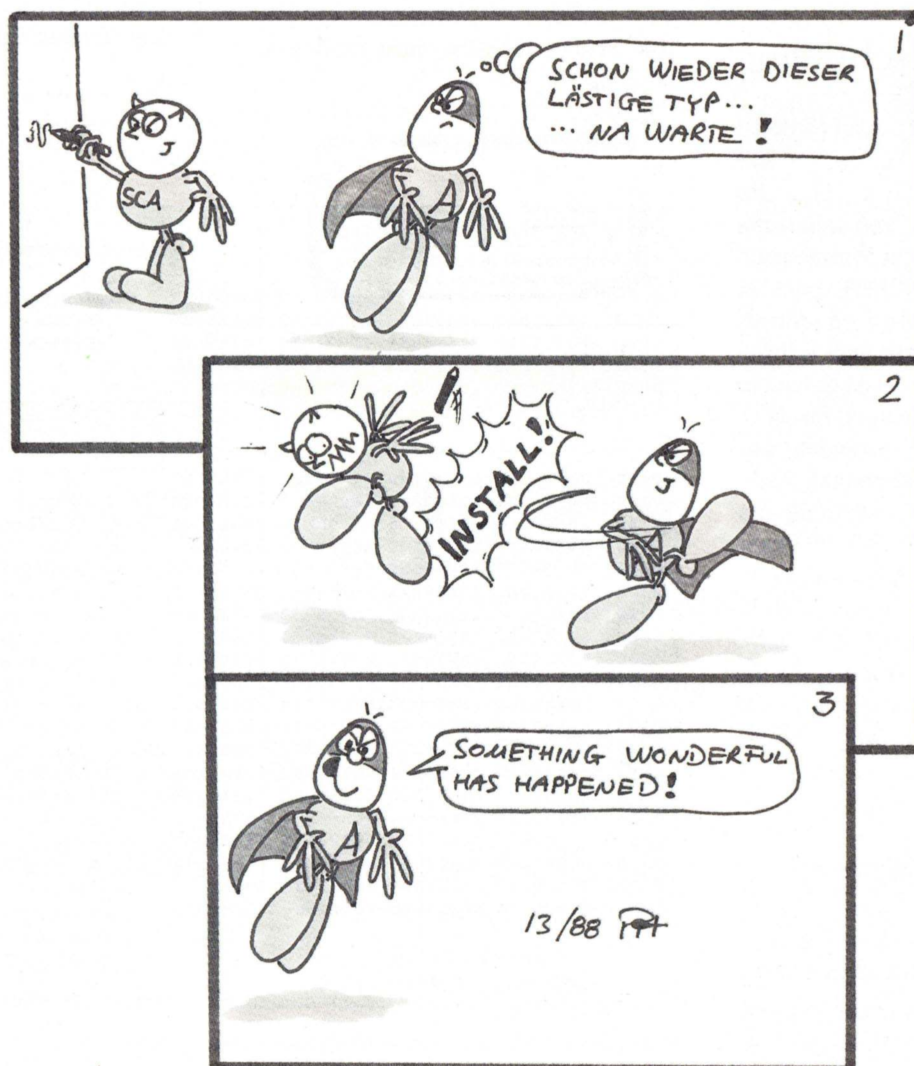
- Reset ausführen (Ctrl/Amiga u. links/ Amiga rechts).
  Wenn die zu überprüfende Diskette infiziert war, wird bei diesem Vorgang der Virus aus dem Speicher auf die im Laufwerk befindliche Diskette geschrieben.

- Erneut einen Reset ausführen. Nun kommt es darauf an: Wenn die zu überprüfende Diskette virusfrei war, ist auch die vorhin formatierte Diskette virusfrei geblieben, und der AMIGA fordert ganz normal zum Einlegen der Workbenchdiskette auf. Wenn aber ein Virus auf der Diskette war, führt der AMIGA einen Warmstart aus, und es erscheint das AMIGA-DOS-Window auf dem Bildschirm mit Copyright-Hinweis und Eingabeprompt. Denn die bekannten Viren DASA und BYTE BANDIT machen die infizierten Disketten auf jeden Fall bootfähig.

Was tun, wenn man einen Virus lokalisiert hat? Die einfachste Möglichkeit, einen virusfreien Bootblock herzustellen, ist der Install-Befehl des Amiga-DOS. Also: Computer ausschalten (mindestens 5 Sek.), mit virusfreier Workbench starten, CLI aufrufen, Befehl "*Install*" aufrufen. Bei zwei Laufwerken ist dies am einfachsten: infizierte Diskette in df1: und Befehl "*Install df1:*" absetzen; bei nur einem Laufwerk zunächst "*Install?*" eingeben, dann Workbenchdiskette raus, infizierte Diskette rein und "*df0:*" eingeben. Die Sache hat allerdings einen kleinen Haken: Es gibt (zum Glück ist das aber die Ausnahme) Disketten mit manipuliertem Bootblock, die nach Anwendung des "*Install*"-Befehls nicht mehr laufen. Also doch lieber vorher noch eine Kopie der infizierten Diskette (die natürlich ebenfalls virusinfiziert sein wird) anfertigen. Die formatierte Diskette kann man natürlich solange für diese Tests weiterverwenden, wie man "saubere" Disketten feststellt, solange also auch die formatierte Diskette saubergeblieben ist. Ich wünsche Ihnen jedenfalls, daß Sie bei allen derartigen Überprüfungen virusfreie Disketten feststellen können.

Natürlich können Sie auch einen Diskettenmonitor zur Virusenttarnung heranziehen. Alle bisher bekannten Viren befinden sich auf der Bootspur (Block 1 u. 2). Alle abgedruckten Abbildungen stellen einen infizierten Bootblock dar. Welchen Virus Sie gefunden haben, können Sie somit schnell feststellen.

Keine Angst vor Viren!



Einkaufsführer

Hier finden Sie Ihren Commodore/Amiga Fachhändler

1000 Berlin



RUNOW
Büroelektronik
Keithstraße 26 · 1000 Berlin 30
☎ 26 111 26

COMPUTER-STUDIO
Schlichting
... die etwas andere Computerei
ATARI-Fachmarkt
MS-DOS Fachmarkt · NEC-Fachhandel
Katzbachstraße 8 · 1000 Berlin 61
☎ 030/786 43 40

2000 Hamburg

Bit Computer Shop
Osterstraße 173 · 2000 Hamburg 20
Telefon: 040/49 44 00

Createam
Computer Hard & Software
Bramfelder Chaussee 300 · 2000 Hamburg 71
Telefon: 040/641 64 73+641 68 61

Computer
Hardware · Software · Zubehör
Lilienstraße 32
(beim Mönckebergbrunnen)
2000 Hamburg 1
Tel. (0 40) 33 67 08

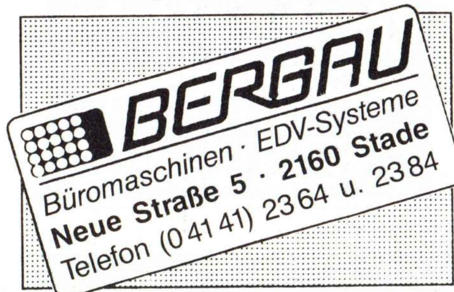


SYSTEMSHOP ®

Computer & Zubehör-Shop
Gerhard u. Bernd Waller GbR
Kieler Straße 623
2000 Hamburg 54
☎ 040/570 60 07
BTX 040 570 52 75

GMA mbH
040/251 41 67/7
Systemhändler
Wandsbeker Chaussee 58
2000 Hamburg 76

2160 Stade



2300 Kiel

Home Computer Laden
Hardware
Software
Service
Spezialisiert auf
Public Domain
Immer die neueste Software auf Lager
Gutenbergstraße 5 · 2300 Kiel · Tel. (04 31) 55 55 55

2390 Flensburg



2900 Oldenburg

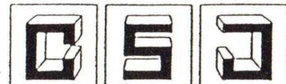
GOLDT
Computerhaus
Donnerschweer Straße 127-129
(gegenüber Weser Ems Halle)
2900 Oldenburg
Telefon (04 41) 88 47 06

2940 Wilhelmshaven

Radio Tiemann
Commodore-Systemfachhändler
Marktstraße 52
2940 Wilhelmshaven
Telefon (0 44 21) 2 61 45

3000 Hannover

COM DATA
Am Schiffgraben 19 · 3000 Hannover 1
Telefon (05 11) 32 67 36



Hannover´s Softwarethek Nr. 1
DIE AUSWAHL an Software für C 16, C 64 / 128, Amiga, Atari ST u. IBM PC.
An der Tiefenriede 27 · 3000 Hannover 1
Tel. 05 11/88 63 83

3500 Kassel

Hermann Fischer GmbH
Commodore-Systemfachhändler
Rudolf-Schwander-Str. 5-13
3500 Kassel
Telefon (05 61) 70 00 00

4500 Osnabrück

B.O.S.S.
Computerhaus
Am Berliner Platz
Goethering 3
4500 Osnabrück
Telefon (05 41) 2 65 70

4650 Gelsenkirchen-Horst

MENTIS GmbH
Hard- und Software, Literatur
Bauteile, Service, Versand
Groß- und Einzelhandel
Poststraße 15 · 4650 Gelsenkirchen-Horst
Telefon (02 09) 5 25 72

6000 Frankfurt



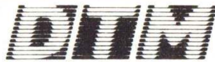
COMPUTER bco

Oederweg 7-9
6000 Frankfurt/Main 1
☎ (069) 55 04 56 - 57

Commodore OKI ATARI TOSHIBA

6200 Wiesbaden

Poststraße 25
6200 Wiesbaden-Bierstadt
Telefon (06121) 560084
Telefax (06121) 563643



Werbung und EDV GmbH

AUTORISIERTER
COMMODORE
SYSTEM-HÄNDLER



Commodore

COMPUTER TREFF

Computerbedarf, PD und Software für ATARI, AMIGA, PC

Nettelbeckstraße 12
6200 Wiesbaden
Tel. (06121) 404302

6380 Bad-Homburg

PDC GmbH
Produkte u. Details Computerverband

Louisenstraße 115
Ladenpassage Alter Bahnhof
6380 Bad-Homburg
Telefon (06172) 24748



Autorisierter Commodore-Systemhändler!

6457 Maintal

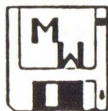
Landolt-Computer
Beratung · Service · Verkauf · Leasing



Autorisierter Commodore-Händler

Wingertstr. 112 · 6457 Maintal/Dörnigheim
Telefon (06181) 45293

6551 Fürfeld



MICHAEL WEISGERBER

HARD + SOFT

Rathausstraße 2
6551 Fürfeld
Telefon (06709) 778

6680 Neunkirchen

SHOP 64
Computer GmbH
Neunkirchen
Telefon (06821) 23713

Commodore Systemhändler

Saarbrücken
Saarlouis
Homburg
St. Ingbert
Neustadt
Trier
Kaiserslautern

6800 Mannheim



Computersysteme + Textsysteme

6800 Mannheim 24
Casterfeldstraße 74-76
Telefon (0621) 850040 · Teletex 6211912

7000 Stuttgart

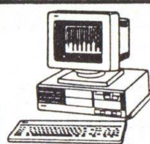
»If AMIGA, go to Schreiber«
Stuttgart's starker Computer-Laden.

SCHREIBER COMPUTER

Rotebühlplatz 10
7000 Stuttgart-1
Tel. 0711/227099

Neu · Neu · Neu · Neu
im SUBWAY
BREUNINGER City

7800 Freiburg



Comp.Z

Pochgasse 31
7800 Freiburg
T. 0761/554280



Commodore

AUTORISIERTER
COMMODORE
SYSTEM-HÄNDLER

7890 Waldshut-Tiengen

hettler-data
service gmbh

Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon (07751) 3094

8000 München

Commodore

AMIGA + PC, HARD- UND SOFTWARE
— BERATUNG / VORFÜHRUNG —
VIDEOBEARBEITUNG

MODL
FOTO · VIDEO · AUDIO · FERNSEH
COMPUTER · GELEGENHEITSMARKT
REPARATUREN · FOTOKOPIEN
FARBKOPIEN · PASSBILDER

MAX-WEBER-PLATZ · 8000 MÜNCHEN 80 · ☎ 4 80 16 50

8400 Regensburg

Zimmermann elektroland

8400 Regensburg
Dr.-Gessler-Str. 8
☎ 0941/95085

8390 Passau
Kohlbruck 2a
☎ 0851/52007

8700 Würzburg

SCHOLL BÜROTEAM

Hardware · Software
Service · Schulung

computer center
am Dominikanerplatz
Ruf (0931) 30808-0

Bei uns werben bringt

GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057

BUF

Schweiz

Computer Trend

Ihr Computer Spezialist

5000 Aarau, Bahnhofstrasse 86,
Tel. 064/22 78 40
4102 Basel-Binningen, Kronenplatz,
Tel. 061/47 88 64
5430 Wettingen, Zentralstrasse 93,
Tel. 056/27 16 60
8400 Winterthur, St. Gallerstrasse 41,
Tel. 052/27 96 96
8021 Zürich, Langstrasse 31,
Tel. 01/241 73 73

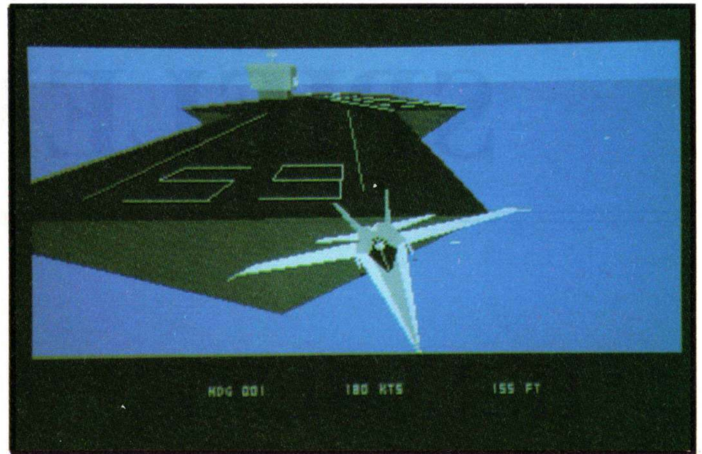
Grösste Auswahl an Peripherie, Software, Literatur und Zubehör.

ATARI

Commodore

TOP 12

1.	(1)	Interceptor



2.	(3)	Ports of Call
3.	(2)	Starglider II
4.	(-)	Elite
5.	(4)	Carrier Command
6.	(10)	Shanghai
7.	(8)	Bards Tale II
8.	(-)	Kartarkis
9.	(9)	Wizball
10.	(5)	Great Gianna Sisters
11.	(7)	Tetris
12.	(12)	Ooze

Schon wieder ... werden die meisten sagen, denn schon wieder sind INTERCEPTOR und PORTS OF CALL auf den vordersten Plätzen, aber so wurde es halt entschieden. Interessant ist der Neueinstieg und gleichzeitige Aufsteiger des Monats ELITE, und auch von den brandneuen Spielen DUNGEON MASTER und DRAGONS LAIR darf man einiges erwarten.

Auch diesmal sollten Sie bei der TOP 12 teilnehmen, denn es gibt wieder 12mal den TOP 12-TIP zu gewinnen. Dies ist in der Regel ein brandneues Spiel, das bei der Redaktion besonderen Gefallen gefunden hat. Die Wahl ist diesmal sehr schwer gefallen, doch nach langem Überlegen haben wir uns für das Rollenspiel DUNGEON MASTER entschieden (siehe Bericht), das endlich auch für den AMIGA erschienen ist. Lassen Sie sich überraschen, was es das nächste Mal sein wird - und vergessen Sie nicht, Ihre Postkarte rechtzeitig einzuwerfen.

TOP 12 TIP

Mitmachen kann jeder, der die TOP 12-Postkarte (bitte keine andere!) ausfüllt und an uns zurückschickt. Der Rechtsweg ist dabei ausgeschlossen. Einsendeschluß ist der 30. Januar 1988. Karten, die nach diesem Termin eintreffen, werden im nächsten Monat berücksichtigt.

Wir gratulieren den Gewinnern dieses Monats, die jeweils einmal den TOP 12-TIP DUNGEON MASTER gewonnen haben: Stefan Pfes*torf, Göttingen / Andreas Minuth, Hameln / Frank Ganady, Essen / Patrick Stein, Weißenthurn / Alexander Neugebauer,* Pforzheim / Bernd Müller, Nordweststadt / Ingo Bernd, Hamburg / Martin Scheidt, Oriedorf / Jörg Wagner, Weilrod / Cary Chant, Harmsdorf / Michael Albers, Cappeln / Zieger Konrad, Stockstadt

DAS

NEUESTE

VOM

SPIELEMARKT

HELLBENT

Im Jahre 2035 versucht Captain Drak Hellbent, die kraelischen Thronräuber aus Aldanicho zu vertreiben. Wen der Hersteller NOVAGEN mit diesem Spiel vertreiben will, war bei Redaktionsschluß nicht bekannt. Außer im Namen unterscheidet sich Hellbent in nichts von mindestens hundert anderen mittelmäßigen "Shot-em-up's", bei denen ein kleines Raumschiff von unten nach oben (und wieder zurück) fliegt, Energie und Schutzschilde eingesammelt und andere kleine Raumschiffe abgeschossen werden müssen.

Die Graphik ist mittelmäßig, der Sound adäquat, der Rest Schweigen.



HUEY oder PFUI?

Wer sich immer noch an Hellbent erinnert (siehe links), der hat es auch verdient, etwas über Huey zu lesen. Nein, nicht wieder das kleine Raumschiff, das ..., diesmal ist es ein Hubschrauber, der Bildschirm scrollt von links nach rechts, und der Hintergrund ist eine Dschungellandschaft. Die Graphik ist akzeptabel, aber alles andere ist absoluter Durchschnitt. Als Minuspunkt kommt noch die sehr gewöhnungsbedürftige Steuerung hinzu.



FOOTBALL DIRECTOR II

"Remove any ext. drives ... loading time 4 mins". D&H GAMES haben sich mal wieder mit einem BASIC-Programm ins Verkaufsgeschäft gestürzt. Ob Football Director II allerdings ein Erfolg wird, hängt davon ab, ob es genug Fußballbegeisterte gibt, die das Programm kaufen, ohne es vorher anzuschauen. Zwar bietet FD II sehr viele Optionen - Spielerbörse, Bankanleihen, Tabellen etc., aber erstens war die uns vorliegende Version nicht fehlerfrei (ohne Schulden ins Menü REPAY MORTAGE), und zweitens ist Football Director II nicht nur zu teuer, sondern auch zu laaaaangsam.

FREEDOM

Kurz vor Redaktionsschluß erreichte uns noch FREEDOM von Coktel Vision. Da es uns leider nicht gelungen ist, das Programm auf einem unserer Rechner zum Laufen zu bringen (das Programm beschränkt sich auf A500 und A2000), wollen wir das Cover für sich selbst sprechen lassen: "...Im achtzehnten Jahrhundert befinden Sie sich auf einer tropischen Plantage einer zuckerproduzierenden Kolonie als Anführer einer Sklavenrebellion und versuchen den heldenhaften Lauf um die Freiheit.

...- *zünden Sie Gebäude und Felder an*
- *fordern Sie Ihre Feinde mit großen Messern heraus*
- *nehmen Sie es mit wilden Doggen auf...*"

und vergessen Sie nicht "...
- *achten Sie dabei auf die Zähler für Leben, Zeit und Verfügbarkeit...*"

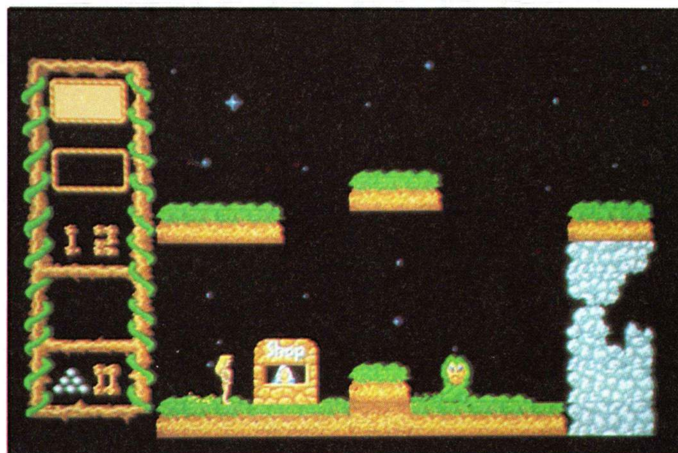
WANTED?

Im Wilden Westen sind die Kopfgeldjäger los. Gesucht werden vier schwere Jungs,auf die bis zu 20000 US$ Kopfgeld stehen. Ausgesetzt hat diese Kopfprämie der Softwarehersteller INFOGAMES. Andererseits ist es sicher nur ein Gerücht, daß man sich die Prämie auch verdienen kann,,wenn man die vier AMIGA-User findet, die sich dieses Spiel kaufen werden. Betrachtet man WANTED genauer, so fällt auf, daß es genauso gut ein Weltraum-Schießspiel sein könnte. Anstelle von kleinen Jetfightern wurden lediglich Coboys eingebaut. Diese stürzen sich dem Revolverhelden entgegen und schießen. Der Held antwortet mit den gleichen Mitteln und kann unterwegs noch zusätzliche Waffen, wie beispielsweise Dynamitstangen, einsammeln. Ansonsten - wie gehabt.



CHUCKIE EGG

Chuckie Egg ist ein Plattform-Spiel, bei dem Spielidee, Sound und Graphik sehr stark an den guten alten 64er erinnern - die Steuerung kann diesem Vorbild allerdings nicht gerecht werden. AMIGA-Besitzer, die keine allzu hohen Ansprüche stellen, werden trotzdem mit Chuckie Egg nicht unzufrieden sein, falls sie nicht mehr als 29.95 DM dafür ausgeben müssen.



VETERAN

Das nach Meinung der Redaktion eindeutig übelste Spiel des Monats war VETERAN von Software Horizons Ltd.. Nachdem man sich zwischen einer Mauser-, Ingram-, Heckler- und Kalaschnikov-Maschinenpistole entschieden hat, geht das Gemetzel los. Ähnlich wie bei POW (allerdings mit schlechterer Graphik) spritzt das Blut nur so über den Bildschirm. Veteran gehört schnellstmöglich auf den Index. All denen, die so etwas trotzdem kaufen, geschieht es recht - es gibt schließlich billigere Möglichkeiten, Leerdisketten zu erwerben.

AUSTRALO PITICUS MECHANICUS

Andreas von Lepel, berühmt-berüchtigter Programmierer von Sarcophaser, ist auf der Suche nach einer neuen Geldquelle. In einem weitbekannten Softwarehaus nahe der holländischen Grenze überfällt sein Hack and Slay Team den Geschäftsführer und nötigt ihn, Australo Piticus Mechanicus zu vertreiben.

APM ist ein typisches Lauf-Hüpf-Aufsammelspiel mit verschiedenen Waffen. Geheimlevel und Rätsel heben den unter der schlampigen Programmierung (kein Bildschirmscrolling, zuwenig Animationsphasen und schlechte Steuerung) leidenden Spielspaß. Die Graphik zeigt gute Ansätze, erscheint insgesamt aber etwas dilettantisch. Von Sebastian Dosch haben wir auch schon Besseres gesehen. Trotzdem kann man sich stundenlang damit beschäftigen - schon weil es so lange dauert, bis man damit zurechtkommt.

DOUBLE DRAGON

'Schlag drauf und Schluß' - ist das Motto dieses Spiels. Ein oder zwei Spieler können sich mit den Bösewichtern prügeln. Eine witzige Idee ist es, daß man diesen dabei ihre Waffen abnehmen kann, um sie sodann gegen sie einzusetzen. Die Graphik ist recht ansprechend, aber wie so oft ist das Titelbild das Beste am ganzen Spiel. Besonders unangenehm berührt hat uns, daß das Programm während des Betriebs mehrmals zielsicher einen Reset auslöste. Trotzdem liegt Double Dragon bei den Neuveröffentlichungen qualitativ in der oberen Hälfte. Dies spricht allerdings eher gegen die jüngst erschienen Titel, als für Double Dragon.



DAS

NEUESTE

VOM

SPIELEMARKT

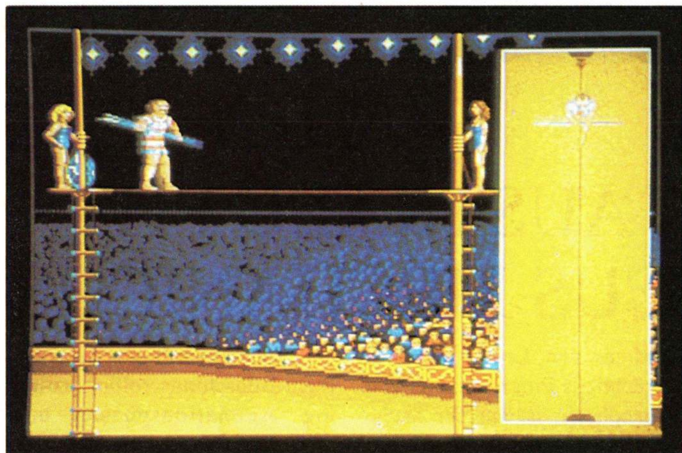
KURZ BELICHTET

Die folgenden Spiele sind erst in allerletzter Sekunde eingetroffen. Wegen der sehr erfreulichen Qualität wollen wir wenigstens kurz darauf hinweisen. Ein ausführlicher Test folgt.

ddf

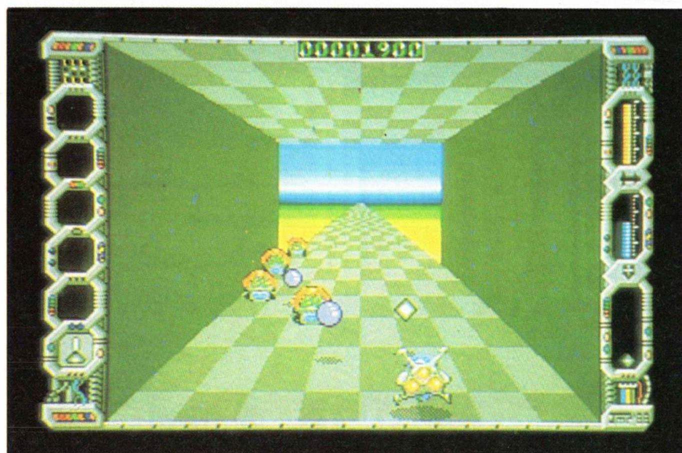
CIRCUS GAMES

von TYNESOFT ist im Stil der "Summer-Winter-Western-etc.-GAMES" gehalten. Als Disziplinen gibt es einen Hochseilakt, Trickreiten, Trapez und Tigertraining. Internationale Wettbewerbe können von bis zu vier Spielern ausgetragen werden. Das Zirkusspiel ist sehr unterhaltsam, auch Graphik und Sound können sich sehen bzw. hören lassen.



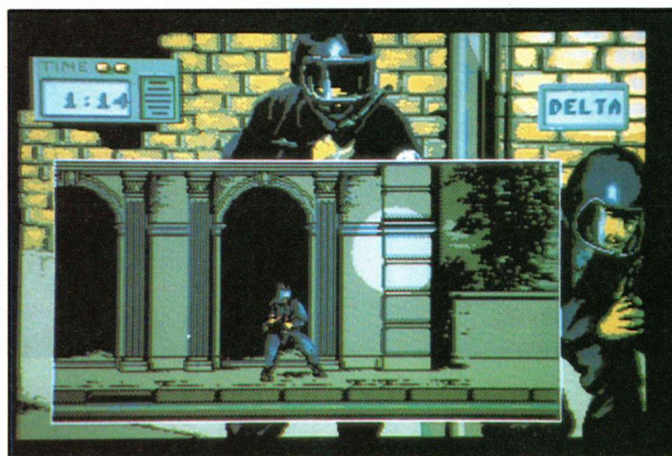
ELIMINATOR

von HEWSON ist ein sehr farbenfrohes, extrem schnelles Schießspiel. Graphik und Sound sind sehr ansehnlich. Mit einem Raumschiff jagt der Spieler eine Piste entlang, wobei er versucht, Gegner abzuschießen und Bonusgegenstände aufzusammeln. Die Spielidee ist nicht neu, aber Spaß macht Eliminator allemal.



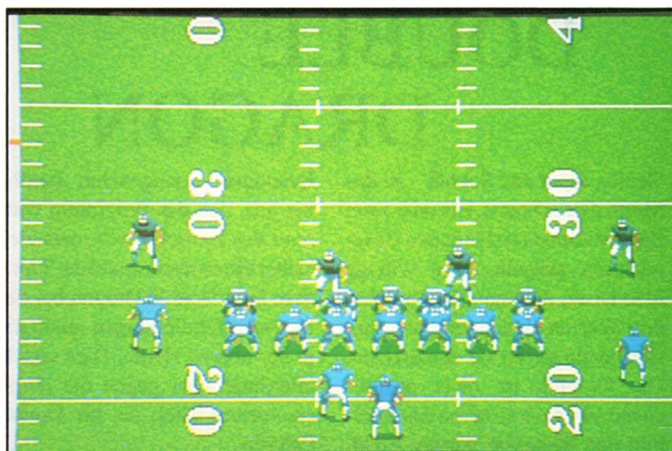
HOSTAGES

von Infogrames: In der Botschaft werden die Angehörigen von Terroristen als Geiseln gehalten. Diese zu befreien,ist das Ziel Ihrer Mission. Dabei kommt es allerdings eher auf Geschicklichkeit als auf "Ballern" an. Graphik, Sound und Animation sind exzellent.



TV SPORTS FOOTBALL

von Cinemaware ist ein American Football-Spiel. Bob Jacob und sein Team werden wieder einmal dem hohen Anspruch gerecht, den die Kunden an das Label CINEMAWARE stellen. Graphik, Animation und Spielmöglichkeiten sind wieder einmal beispielhaft für den AMIGA. Wenn es mehr Produkte dieser Qualität gäbe, müßte Commodore keine Angst vor Spielekonsolen haben.



SOLDIER OF LIGHT



Soldier of Light läßt den Joystick nicht kalt werden, Actionfans kommen auf ihre Kosten.

Eindringlinge aus einer anderen Galaxis haben drei Planeten unserer Milchstraße überfallen und ein ganzes Waffenarsenal abgeladen, um für einen Rückeroberungsversuch gewappnet zu sein. Laserfahrzeuge, Kanonen und ein Haufen Soldaten mit Spezialwaffen patrouillieren und haben die Anweisung, jeden Eindringling zu vernichten.

Es hilft alles nichts! Jemand muß auf die drei Planeten und den Bösewichten zeigen, wer Herr der Galaxie ist. Das geht natürlich nicht mit Diplomaten und Aktenköfferchen. Hier hilft nur einer: Xain, der Superheld, der ausgerüstet mit Jetstiefeln und einer Laserkanone die Eindringlinge vertreiben soll. Per Joystick stürzen Sie sich ins Abenteuer. Zunächst erscheint eine Art Menü auf dem Bildschirm. Mit einem Raumschiff sollen Sie den Planeten anfliegen, den Sie als erstes zu säubern gedenken. Kaum hat man sich einen Planeten ausgesucht, geht die Action auch schon los: Ein Raumschiff schwebt über der Planetenoberfläche, und Xain springt heraus. Sofort greifen die feindlichen Soldaten an. Xain läuft bei horizontalem Scrolling über die Planetenlandschaft und ballert mit seiner Laserkanone auf die Soldaten, Geschützfahrzeuge und Laserkanonen, die ihn von allen Seiten angreifen. Die Soldaten und Xain selber sind in Raumanzüge gehüllt und haben sehr viel Ähnlichkeit mit Menschen.

Mit Hilfe seiner Jetstiefel vermag es Xain, in ein anderes Stockwerk oder auf in der Luft fliegende Felsen zu schweben. Sie müssen immer geradeaus laufen, Extrawaffen einsammeln und natürlich ballern, ballern und nochmals ballern. Sind Sie durch die gesamte Landschaft eines Planeten gelaufen, wartet ein gefährliches Monster auf den Helden. Jetzt sind äußerste Konzentration und schnelle Reaktionen gefragt, denn die Monster sind eine echt harte Nuß. Sollte es Ihnen gelingen, ein Monster zu besiegen, gibt es eine kleine Shoot'em up-Szene, in der Sie ein Raumschiff bei horizontalem Scrolling durch den Weltraum fliegen.

Dann geht es auf den nächsten Planeten. Ein ähnliches Gemetzel beginnt von vorne. Hat man auch den dritten Planeten durchgespielt, muß sich der Spieler beim Todesstern beweisen. Manchmal sind die Alienangriffe ein bißchen heftig, was unweigerlich zum Verlust eines Bildschirmlebens führt. Doch mit etwas Übung entkommt man auch aus noch so aussichtslos erscheinenden Situationen. Im Grunde genommen ist Soldier of Light eine neue Ballervariante, nur muß der Spieler diesmal kein Raumschiff steuern, sondern den Helden. Die guten Grafiken und die nervenaufreibende Action werden Actionsfans lange Zeit in Atem halten.

CBO

Hersteller: Ace



SPEEDBALL

IMAGE WORKS zeichnet sich verantwortlich für eine eigenwillige Sportsimulation. Eine Mischung aus Rugby, Handball und der vom Film her bekannten Sportart Rollerball. Die Zeiten ändern sich so auch die Sportarten. In ferner Zukunft wird nur noch SPEEDBALL gespielt. Eine Mannschaftssportart, in der fünf hartgesottene Akteure in einer geschlossenen Arena einen Energieball im gegnerischen Tor unterbringen müssen. Große Regelbeschränkungen gibt es hierbei nicht, gemeine Tricks sind an der Tagesordnung.



SPEEDBALL ist die Sportart des 2100 Jahrhunderts, stahlharte und skrupellose Akteure sind die Helden der Spielarenen.

Zu Beginn von SPEEDBALL kann zwischen mehreren Optionen gewählt werden. Beispielsweise kann die Anzahl der Spielrunden oder der Spieler definiert werden. Spielstände können eingeladen bzw. abgespeichert werden, was Sie mit der Zeit zu schätzen wissen. Bevor es in die Spielarena geht, kann zwischen drei Charakteren gewählt werden, die sich in Größe, Gewicht und einigen weiteren Punkten voneinander unterscheiden. Das Spielfeld sieht der Spieler aus der Vogelperspektive, wobei nur ein Teil des Feldes sichtbar ist, das, je nachdem nach oben bzw. unten gescrollt wird. Die Grafik ist ausgezeichnet und sehr schnell animiert.

Die einzelnen Mitglieder einer Mannschaft können Pässe werfen, Sololäufe veranstalten und gezielte Würfe auf das Tor unternehmen. Die Spielfigur, die gesteuert werden kann, ist weis umrandet, hat sie den Energieball, färbt sich der Rand rot bzw. grün. Um an des Gegners Ball zu gelangen, stehen unfeine Mittel zur Verfügung, ein unsanfter Rempler läßt den Gegner schon mal Bekanntschaft mit dem Boden machen, und man ist im Besitz der Kugel. Sprünge in des Gegners Rücken oder, um eine hochgeworfene Energiekugel zu fangen, geben SPEEDBALL eine weitere interessante Variante. Die Spielarenen ändern sich von Spiel zu Spiel nur minimal, lediglich Ablenkkugeln wechseln ihren Standort. Mitunter tauchen auf dem Spielfeld Buchstaben oder Energiepunkte auf, die eingesammelt werden können und der Mannschaft einige Vorteile bringen.

Was sich an Action und guter Unterhaltung mit Speedball dem AMIGA-Spieler bietet, ist bisher von Sportspielen noch nicht dagewesen. Keine ruhige Minute oder besser Sekunde können sich die Spieler gönnen, ohne ins Hintertreffen zu geraten. Besonders das Spielen zu zweit macht einen Heidenspaß, wer möchte seinem besten Freund nicht mal eins auswischen?!



erscheint der Schwachpunkt des Todessterns: der Reaktor! Mit einem gezielten Schuß wird dieser zerstört. Anschließend heißt es: Kehrt marsch! Denn der Reaktor explodiert. Hat man die Stahlbauten heil hinter sich gelassen, fliegt das Raumschiff in den Weltraum, und der Todesstern explodiert. Das Spiel beginnt nun mit höherem Schwierigkeitsgrad von vorne. Insgesamt gesehen sind bei "Return of the Jedi" wesentlich mehr Elemente des Films eingeflossen als bei den Vorgängern. Ausgefüllte, farbenfrohe Grafik, die dazu noch schnell ist, lassen das Game positiv erscheinen. Die aufeinander aufbauenden Schwierigkeitsgrade steigern die Motivation ebenfalls. Ein Anfänger muß nicht gleich Chewbacca im Geher und Lando in Falken steuern. Dennoch bleiben einige Kritikpunkte zu erwähnen: Jemand, der von den Filmen noch nie etwas gehört hat, wird mit dem Game nicht viel anfangen können, da kein Zusammenhang zwischen den einzelnen Spielstufen besteht. Sieht man einmal von den Übersetzungsschwächen der Anleitung ab, handelt es sich bei "Return of the Jedi" handelt es sich durchaus um ein gelungenes Game, das sowohl grafisch als auch spielerisch besser ist als die beiden ersten Spiele der Star Wars-Trilogie. CBO

Hersteller: Domark



SWORD OF SODAN

Nur wenige Programme wurden mit so viel Vorschußlorbeeren versehen wie SWORD OF SODAN. Bereits vor einem Jahr konnten die ersten Demos bestaunt werden. Das Besondere war und ist die Programmierung und die Animation von bildschirmgroßen Charakteren. Der AMIGA macht's möglich, durch die sogenannten BOBS. Die endgültige Version des Spiels lag uns nun vor. Was die Programmierer von Discovery Software mit diesem Produkt geleistet haben, ist anerkennenswert und sucht seinesgleichen auf den AMIGA. Was an Grafik, Sound und Animation geboten wird, ist Spitzenklasse. Die Abwechslung kommt hingegen etwas kurz.





Auf dem Friedhof liegen die Toten leider nicht ruhig im Grab.

SWORD OF SODAN spielt in einer typischen Fantasywelt mit Dämonen, Monstern, Zauberern, holden Jungfrauen und natürlich Helden und Heldinnen. Brodan und Shardan sind die mutigen Schwertkämpfer und ziehen aus, um den bösen Zauberer Zoras zur Hölle zu schicken und so den Frieden über das tyrannisierte Land zu bringen. Doch Zoras' Schergen sind zahlreich, und seine Burg gleicht einer Festung. Doch Helden kennen keine Furcht.

Insgesamt sind 11 Levels zu bewältigen, die sich zum Teil erheblich voneinander unterscheiden. Zunächst gilt es, die Wachen der Stadtmauer niederzumetzeln, bevor es in der Stadt zu weiteren unliebsamen Überraschungen kommt. Rollende Fässer, wilde Barbaren und ein einen um zwei Haupteslängen überragender Unhold wollen einem den Garaus machen. Der Weg führt weiter durch dichten Wald. Räuber und riesige skorpionähnliche, grüne Untiere hindern den Helden am Weiterkommen. Ist der Friedhof noch überwunden, erreicht der Held schließlich das Foyer der Burg. Doch ab hier erwarten ihn noch weit schlimmere Untertanen des bösen Zauberers. Des Zauberers Lehrlinge probieren ihre magischen Fähigkeiten nur zu gerne an dem unliebsamen Eindringling aus. In den tiefen und dunklen Verliesgängen wimmelt es des weiteren nur so von Fallgruben, herunterstürzenden Decken, Feuerbällen, Säuretropfen, Lavaseen uvm.. Ist auch diese Hürde genommen, trifft man mehr oder minder zufällig auf ein braves Reittier, ohne dessen Hilfe es kein Weiterkommen gibt. Hat der Held endlich den Aufgang zum Burgturm gefunden, ist Zoras aber noch lange nicht erreicht. Ein Kampfungeheuer, größer und stärker als alles Bisherige, kennt kein Erbarmen. Mit etwas Glück und Geschick ist aber auch hier ein Ausweg zu finden, und der unerschrockene Held tritt zur Apokalypse an, tritt Zoras gegenüber und....

SWORD OF SODAN glänzt durch ausgezeichnete Grafik, mit viel Liebe zum Detail. Der zum Teil digitalisierte Sound hinkt der Grafik nur wenig hinterher. Ein Leckerbissen in Grafik und Sound ist beispielsweise das Friedhofslevel. Sie werden begeistert sein. Um den Helden nicht ganz alleine zu lassen, gibt es zwei Bonusobjekte, die bei Aktivierung den unmittelbaren Gegner vernichten oder den Helden für 30 Sekunden mit einem magischen Schild versehen. Letzteres benötigen Sie unbedingt für die Überwindung des Kampfungeheuers. In der linken oberen Ecke des Bildschirms können Sie die Lebensenergie und die momentan abrufbereiten Bonusobjekte ablesen.

Zu Beginn des Spiels können Sie zwischen zwei Charakteren wählen. Wenn Sie einen AMIGA mit mehr als 512 KByte besitzen, können Sie sich des weiteren eine Wiederholung des letzten Spieles ansehen, bis zu zwei Stunden versprechen die Programmierer. Negativ fielen bei SWORD OF SODAN allerdings auch ein paar Dinge auf: Das Doppelscrolling von Hinter- und Vordergrund könnte etwas ruckfreier sein, auch sind einige Szenen recht makaber und brutal. Für den einen oder anderen ist die eingeschränkte Handlung noch ein Manko, schließlich muß der Spieler fast nur mit dem Schwert draufloshauen. SWORD OF SODAN wird trotzdem seine Anhänger finden.

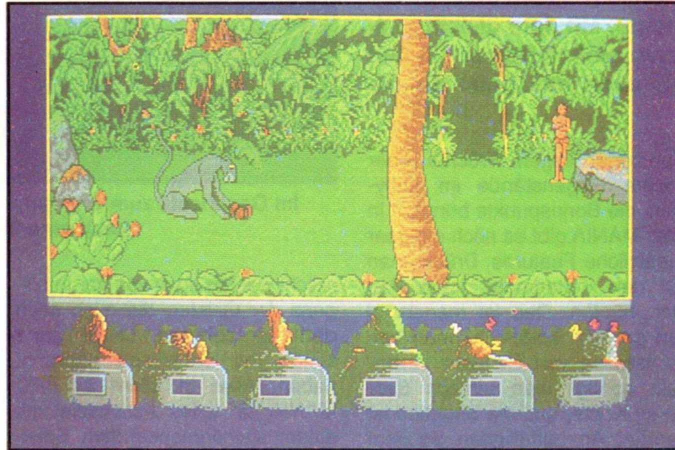
AK

Anbieter: CDC, Bad Homburg



DAS DSCHUNGELBUCH

Walt Disneys Zeichentrickfilm "*Das Dschungelbuch*" war ein Riesenerfolg in den Kinos. Vor kurzem erschien die "Versoftung" des französischen Softwarehauses "Coktel Vision". Die Handlung ist natürlich die gleiche wie im Film: Mowgli lebt seit seiner Geburt im Dschungel. Er liebt das Dschungelleben und möchte es auf keinen Fall missen. Doch ein Typ namens Bagheera will Mowgli in ein Dorf zu den Menschen bringen. Dieser hat jedoch auf Zivilisation gar keinen Bock. Deshalb flieht er in den Dschungel, klettert auf Tempelanlagen, versteckt sich in Höhlen und läuft durch die Savanne. Am Ende seiner Flucht soll Mowgli ein hübsches Mädchen treffen, das zu ihm paßt. Doch der Weg ist weit und beschwerlich. Auf seiner Reise trifft der Held viele Tiere. Einige sind ihm wohlgesonnen, andere nicht. Sie versperren Mowgli den Weg oder töten ihn sogar. Im Stil der alten "Ultimate Play the Game"-Spiele präsentiert sich die Grafik dreidimensional auf dem Bildschirm. In jedem Bild gibt es Wege zur nächsten Szene. Mit auf dem Boden liegenden Gegenständen wie Kokosnüssen oder Bananen kann der Held werfen oder Tiere weglocken. Jedes Tier, das Mowgli feindlich gesinnt ist, kann nur durch einen bestimmten Gegenstand beseitigt werden.



Das Dschungelbuch faszinierte viele Walt Disney-Fans auf der Kinoleinwand, jetzt kann der kleine Mowgli auch über den Computerbildschirm gesteuert werden.

Das gesamte Geschehen von "Dschungelbuch" spielt sich auf einer Leinwand ab. Im Bildvordergrund sind Kinozuschauer zu sehen. Immer, wenn Mowgli einen Fehler macht, schläft einer der Zuschauer ein. Ist das gesamte Publikum dem Schlafe verfallen, ist das Spiel beendet. Grafisch handelt es sich bei "Dschungelbuch", wie bei vielen Cokteispielen, um einen Leckerbissen. Mowgli ist lustig animiert, genauso, wie wir ihn aus dem Zeichentrickfilm kennen. Spielerisch konnte mich das Produkt leider nicht überzeugen. Tiere mit Kokosnüssen oder Bananen zu bewerfen, ist im ersten Moment ja noch ganz lustig, doch schon nach einer Spielstunde fallen einem vor Langeweile die Augen zu.

CBO

Info: Bomico

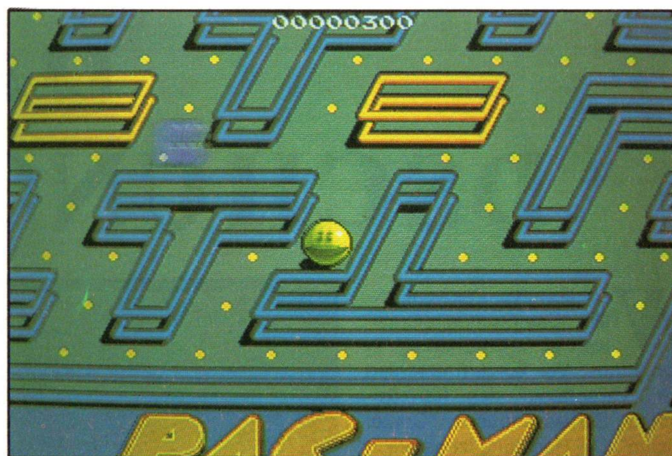


Return of the Jedi

In diesem neuen Programm von Domark haben Sie zu Beginn die Möglichkeit, zwischen drei verschiedenen Schwierigkeitsgraden zu wählen. Dann beginnt die Spielhandlung, die sich eng an der des Films orientiert. Für diejenigen, die den Leinwandknüller verpaßt haben, gebe ich nochmal einen Einstieg in die Geschichte: Der böse Imperator hat einen Todesstern bauen lassen, um den aufsässigen Rebellen den Garaus zu machen. Diese mächtige Waffe, die sogar Planeten zerstören kann, muß vernichtet werden. Spione haben ausgekundschaftet, daß die berüchtigte Kampfstation einen Schwachpunkt hat. Luke Skywalker, Prinzessin Leia und Han Solo landen auf dem Mond Endor, um die Mission zu erledigen. An dieser Stelle kommen Sie ins Spiel: Auf einem Speedbike, einer Art Luftkissenmotorrad, rast Leia bei diagonalem Scrolling durch den Wald, gefolgt von den Soldaten des Imperiums. Mit dem Joystick wird das Speedbike gelenkt. Per Knopfdruck feuert die Bugkanone Schüsse ab. Durch Verdrängen oder Abschießen der Feinde muß sich der Weg freigekämpft werden. Dabei gilt es, Bäumen und Sträuchern auszuweichen. Bei der rasanten Fahrt wird der Spieler von den kleinen, puscheligen Ewoks unterstützt, die Seile spannen oder Baumstammfallen errichten, um die Verfolger aufzuhalten. In höheren Leveln steuert der Spieler einen Geher, der von dem gutmütigen Chewbacca gekapert wurde. Dieser tapst ebenfalls bei diagonalem Scrolling durch den Wald und muß dabei rollenden Baumstämmen ausweichen, die abgeschossen werden können. Zwischendurch wechselt das Szenario: Lando Calrissian wird durch Sie im Rasenden Falken gesteuert. Dieser hat nun die Aufgabe, feindliche Jäger zu eliminieren. Danach geht es wieder auf den Waldmond, wo Chewbacca erneut Baumstämmen ausweichen muß. Zu guter Letzt fliegt Lando im Raumschiff seines Freundes Han Solo in das Innere des Todessterns. Hier werden Sie nun von Jagdschiffen verfolgt, die genau wie auf dem Mond durch Schüsse oder Abdrängen erledigt werden können. Als Hindernisse müssen diesmal nicht Bäume, sondern Stahlstreben herhalten. Nach geraumer Zeit



PAC MANIA
PAC MAN lebt!



Im Gegensatz zum original PAC MAN besticht PAC MANIA durch eine 3D-Darstellung.

PAC MAN ist einer der erfolgreichsten Charaktere in der Computer- und Videospielgeschichte. Mit dem gleichnamigen Automaten machte ATARI einen großen Reibach in den Spielhallen, sehr erfolgreich war auch das Spiel für die ATARI VCS-Konsole.

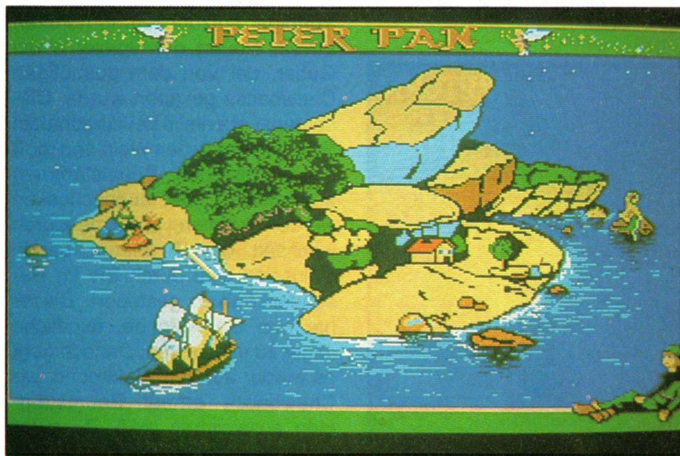
In sämtlichen Variationen gibt es den kleinen, gelben Punktefresser auch auf allen Homecomputersystemen. Dann wurde es eine ganze Zeit lang still um Mr. PAC MAN. Jetzt feiert er in "PAC MANIA" ein ruhmvolles Comeback. Genau wie beim original Pac Man muß der Spieler den Helden durch Labyrinthe steuern und alle gelben Punkte auffressen, um in das nächste Labyrinth gelangen zu können. Dabei wird er von Geistern verfolgt, deren Berührung für den Verlust eines Bildschirmlebens sorgt. Durch das Auffressen von Kraftpillen kann PAC MAN die Geister fressen. Manchmal liegen Früchte oder andere Gegenstände im Labyrinth, die Bonuspunkte bringen. In PAC MANIA gibt es noch ein paar zusätzliche Features. Drückt man auf den Feuerknopf, kann Pac Man hüpfen. Das ist manchmal von großem Vorteil. Ist man beispielsweise von Geistern umzingelt, springt man einfach über deren Köpfe, und die Situation ist gerettet. Am auffälligden. So zieht sich die Action bei der AMIGA-Version über den ganzen Bildschirm, und die Geister sind vortrefflich animiert. So rollen sie beispielsweise mit den Augen, wenn man sie überspringt. Positiv überrascht war ich auch von der Intelligenz der Geister, die mich nach dem fünfzehnten Level ganz schön ins Schwitzen gebracht haben. Alles in allem ist PAC MANIA ein tolles Actionspiel mit einem süchtigmachendem Spielprinzip, das über ausgezeichnete Grafik verfügt. Die Soundeffekte sind leider nicht sonderlich gelungen. Trotz der Schwächen beim Sound kann ich Ihnen PAC MANIA empfehlen, PAC MAN-Fans kommen auf ihre Kosten. Holen Sie sich PAC MANIA, Sie werden es mit Sicherheit nicht bereuen!

CBO

Info: Leisuresoft



Peter Pan
Ein Spiel für "die Kleinen"?



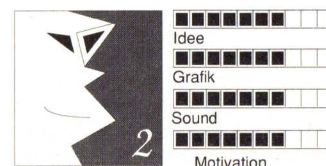
Peter Pan hat ansprechende Grafik und eine Sprachausgabe, die sich hören lassen kann.

Die meisten Computerspiele sind für kleine Kinder nicht geeignet: Strippokerspiele, Metzelspiele oder Strategiespiele, in denen man den zweiten Weltkrieg nachspielen muß, könnten die ABC-Schützen verwirren oder vielleicht sogar "sozialethisch desorientieren".

Trotzdem braucht Papi den Computer nicht vorm kleinen Sohnemann wegzuschließen. Denn der französische Hersteller Coktel Vision eröffnet mit "Peter Pan" eine Serie mit Computerspielen für die kleinen Joystickakrobaten. Klar, daß das Spiel, das übrigens für Kinder zwischen fünf und acht Jahren sein soll, entsprechend dem Alter der Spieler einen geringen Schwierigkeitsgrad besitzt. Für einen geübten Spieler ist es überhaupt keine große Tat, das Game innerhalb von fünf Minuten durchzuspielen. Der junge Spieler wird sich vermutlich längere Zeit damit beschäftigen können. Gleich nach dem Dudeln der Titelmelodie fragt eine digitalisierte Frauenstimme, mit welchem Schwierigkeitsgrad begonnen werden soll. Die Schwierigkeitsstufen sind nach Tiernamen benannt, wobei nach der Wahl des Vogels das Spiel am leichtesten zu bewältigen ist. Jede Aufgabe wird dem Kind in einer klar verständlichen Sprachausgabe dargelegt. Im ersten Spielabschnitt gilt es, einen Schatten zu finden. Man sieht ein Zimmer, in dem man mit der Maus alle Gegenstände anklicken kann. Hat man den Schatten gefunden, schreit der AMIGA "Yippii", das Fenster öffnet sich, und Peter Pan fliegt mit einem fliegenden Teppich zur nächsten Aufgabe. Jetzt sieht das Kind eine Insel. Blinkende Objekte wie Indianerzelte oder Bäume müssen angeklickt werden, um in den nächsten Spielabschnitt zu gelangen. In einem Labyrinth heißt es, Blumen einzusammeln. Jetzt steuert man Peter Pan per Tastatur. Hat man alle Blumen aufgesammelt, muß man im nächsten Bild alle versteckten Nixen mit der Maus anklicken, bevor in einem weiteren Labyrinth ein hübsches Mädchen befreit werden soll. Bis auf die Labyrinthabschnitte sind alle Spielabschnitte in einer ansprechenden, farbenprächtigen Grafik dargestellt. Die Sprachausgabe ist ebenfalls Sonderklasse. Eine derart gut verständliche Sprache habe ich bisher auf dem AMIGA noch nicht gehört. Für meine Begriffe ist Peter Pan viel zu leicht, aber ich bin ja auch nicht mehr acht.

CBO

Hersteller: Coktel Vision



(Achtung! Geeignet für Kinder zwischen fünf und acht Jahren)

OUT RUN

Wollten Sie nicht schon immer einmal einen Ferrari Testarossa fahren? In OUT RUN von U.S. Gold haben Sie die Möglichkeit dazu. Mit einem hübschen Mädel an der Seite rast man unter Zeitdruck über einen amerikanischen Highway und überholt Käfer, Lastwagen und andere Verkehrsteilnehmer.

In der Spielhalle wurde OUT RUN zum Spiel des Jahres gekürt. Nach den ersten schlappen Adaptionen des Automatenknüllers glaubte kaum einer daran, noch einmal eine vernünftige Heimcomputerversion dieses süchtigmachenden Autorennens in die Hände zu bekommen. Amstrad und C64-Version waren im Vergleich zur Spielhallenfassung ein Totalausfall. Natürlich ist die neue AMIGA-Version dem Automaten ebenfalls nicht ebenbürtig. U.S Gold-Programmiererteam Probe Software ist jedoch mit der AMIGA-Version eine passable Umsetzung gelungen. Die Animation der Grafik ist zwar etwas ruckhaft, kann aber im großen und ganzen überzeugen. Soundmäßig ist bei der AMIGA-Version bis auf Kleinigkeiten ebenfalls alles in Ordnung. Satte Klänge klingen dem Spieler aus dem Lautsprecher des Monitors entgegen. Nach dem Laden des Titelbildes ertönt eine Fanfare mit Sprachausgabe und digitalisierten Motorengeräuschen. Prädikat: Hörenswert!



Ein Ferrari und ein hübsches Mädel, was will man mehr?

Im allgemeinen verfügt das AMIGA-OUT RUN über dieselben Features wie das Vorbild aus der Spielhalle. Lediglich das Radio zur Einstellung verschiedener Hintergrundmusiken fehlt. Die Auswahl der Sounds wie der anderen Spieloptionen erfolgt über Pull-Down-Menüs. Wem die Sounds, die das Programm bietet, nicht reichen, der kann das neben Dokumentation und Spieldiskette mitgelieferte Audiotape in den Kassettenrekorder stecken und sich vom Originalautomatensound (Auszeichnung der besten Automatenmusik) berieseln lassen.

CBO

Info: Rushware



Die Arche des Captain Blood

Bei einer Reise durch den Hyperraum passiert es: Die Materie spielt verrückt. Captain Bloods Körper spaltet sich in viele genetische Kopien, die mit hoher Geschwindigkeit in die unendlichen Weiten des Weltraums fliegen.



Ausgezeichnete Grafik, guter Sound und eine neue Spielidee zeichnen "Die Arche des Captain Blood" aus.

Captain Blood muß unbedingt alle Kopien finden, sonst erleidet er einen grauenvollen Tod. Also hüpft er in seine Arche und begibt sich auf eine lange Suche. Nach 800 Jahren hat er fast alle Kopien gefunden, nur fünf sind einfach nicht aufzutreiben. Inzwischen ist Blood ein steinalter Mann, dessen Hände aussehen, wie die eines Skelettes. Klar, daß Blood in diesem hohen Alter nicht alleine durch den Weltraum schippern kann. Als Käufer von Infogrames neuestem Game für den AMIGA "Die Arche des Captain Blood" sind Sie mit dabei. Mit der Hand an der AMIGAmaus steuern Sie Bloods Knochenarm und können so verschiedene Knöpfe im Cockpit der Arche bedienen.

Vielen Planeten müssen Besuche abgestattet werden, um Anhaltspunkte für den Aufenthaltsort der fünf Kopien zu bekommen. Ein Drohnengefährt dient zum Besuch der vielen Planeten. Die Landschaft des Planeten setzt sich aus blauer Fraktalgrafik zusammen, die schnell animiert über den Bildschirm zischt. Durch die vielen Berge und Täler entsteht eine tolle 3D-Perspektive. Ziel jedes Planetenanfluges ist es, einen Bewohner aufzuspüren, der einem wichtige Hinweise gibt. Meist befinden sich die außerirdischen Tratschweiber am Ende eines Tunnels. Dort baut sich der Außerirdische vor dem Cockpitfenster auf, und der Übersetzungscomputer wird aktiviert. Klar, daß die Außerirdischen kein Deutsch können. Doch Dank der Technik ist es möglich, mit den Typen zu kommunizieren. Die Außerirdischen quatschen in unverständlichen Worten, die auch in Bildsymbolen auf dem Bildschirm erscheinen. Wollen Sie dem Burschen antworten, wählen Sie per Maus auf einer von links nach rechts scrollenden Leiste die entsprechenden Symbole dazu aus. Jedes Symbol steht für ein Nomen, Verb oder einen anderen Satzbaustein. Mit knapp 100 verschiedenen Sprachsymbolen lassen sich zwar keine philosophischen Diskussionen führen, aber für das Nötigste reicht's. Wichtig ist, daß man den Außerirdischen stets freundlich gesinnt ist. Also, keine Beleidigungen loslassen. Manchmal müssen Sie einem Außerirdischen noch einen Gefallen tun, bevor er mit der Information herausrückt.

"Die Arche des Captain Blood" ist ein Programm mit hervorragender Grafik, einer hörenswerten Titelmelodie und tollen Soundeffekten. Für die Titelmelodie zeichnet der französische Musiker Jean Michel Jarre verantwortlich. Action- und Adventureelemente sind gelungen miteinander verknüpft, jedoch ist "Captain Blood" nicht makellos: Meiner Meinung nach wird das Spiel nach einigen Stunden Spielens langweilig, da sich alle Aktionen ständig wiederholen.

CBO

Info: Bomico

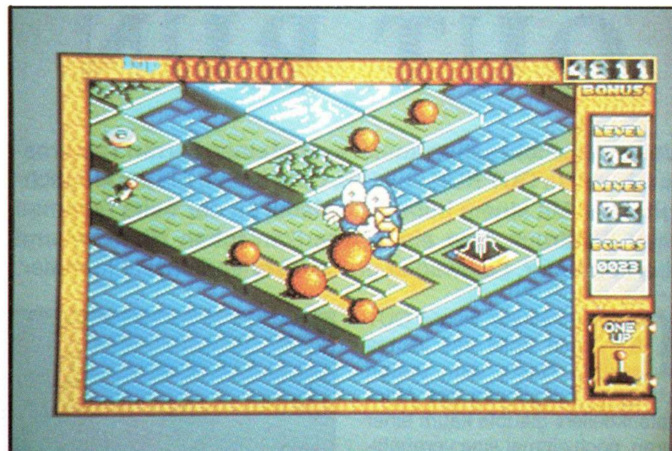


BOMBUZAL

Eine gute Spielidee liegt BOMBUZAL von IMAGE WORKS zugrunde. Nicht wie sonst üblich gilt es, den Joystick und dessen Feuerknopf bis an die Grenzen des Materials zu belasten, sondern bei BOMBUZAL muß auch das Gehirn eingeschaltet werden.

BOMBUZAL nennt sich ein kleines, rundes Monster, dessen Lebensaufgabe darin besteht, Bomben zu entschärfen. Auf den im Laufe des Spiels komplexer werdenden Inseln, die sich aus vielen kleinen Plattformen zusammensetzen, sind drei Größen von Bomben zu finden, die natürlich verschiedene Sprengkraft besitzen. Die kleinste Bombe zündet nur vier Felder, die nächst größere schon 12 usw..

Liegt eine Bombe auf einem benachbartem Feld, wird diese unweigerlich auch gezündet, so kann es schnell zu einer Kettenreaktion kommen. BOMBUZAL kann die Bomben an strategisch günstige Stellen plazieren. Dadurch kann er gezielte Kettenreaktionen auslösen. Viel Zeit bleibt aber nicht, schnelles Denken und Handeln ist gefragt. BOMBUZAL kann entweder in 2D- oder 3D-Ansicht gespielt werden, durch Drücken der F1-Taste kann die Ansicht jederzeit gewechselt werden. 100 Level gilt es zu entschärfen, wobei jedes Level mit einem Codewort versehen ist, so daß der Spieler durch die Eingabe des Codewortes in das betreffende Level springen kann. Zahlreiche weitere Features machen es dem kleinen Monster auch nicht leichter, spiegelglatte oder zusammenbrechende Felder sind nur mit viel Aufmerksamkeit und Geschick zu bewältigen. Die Grafik von BOMBUZAL ist gehobener AMIGA-Standard, obwohl einige Kleinigkeiten stören. Die 3D-Ansicht ist fast unspielbar, viel Übung ist hier vonnöten. Der Sound ist für AMIGA-Verhältnisse etwas zu dürftig geraten. Auch



BOMBUZAL hat es nicht leicht, die vielen Bomben zu entschärfen.

haben die Programmierer einige logische Denkfehler begangen. So kann es mitunter vorkommen, daß ein Level nicht zu schaffen ist, weil einige Hindernisse schlecht plaziert wurden. Im allgemeinen zählt BOMBUZAL aber zu den guten Spielen. Auch auf Dauer wird BOMBUZAL nicht langweilig. Wer kniffelige Denksportaufgaben mag, ist mit diesem Produkt auf jeden Fall gut bedient.



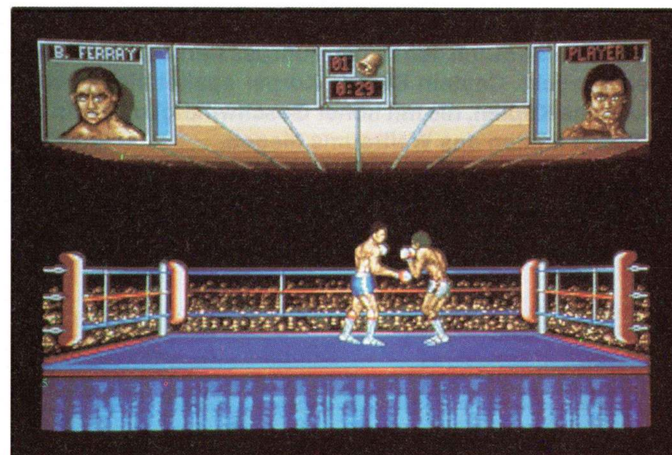
AK

RINGSIDE

BOXEN besitzt für viele eine Faszination, für andere ist es abstoßend und brutal. Trotz der konträren Lager ist eine brandneue Boxsimulation für den AMIGA erhältlich - RINGSIDE.

RINGSIDE wird auf zwei Disketten geliefert, AMIGA-Besitzern mit nur einem Laufwerk steht häufiges Wechseln bevor. Die Titelgrafik des Spiels läßt im grafischen Bereich einiges erwarten. Bevor es aber in den Ring geht, müssen der Name, das Gewicht und das Alter des Boxers eingegeben werden. Zusätzlich kann die Schlagstärke der verschiedenen Grundschläge verändert werden. Es steht aber nur ein bestimmtes Potential zur Verfügung, das, geschickt verteilt, den Erfolg oder Mißerfolg eines Kampfes beeinflußt. Das Potential wächst bei einem gewonnenen Kampf. Die Länge und die Anzahl der Runden können ebenso festgelegt werden wie der Schwierigkeitsgrad. 10 Boxer stehen als Gegner bereit, eine Rangliste gibt Aufschluß über die momentane Plazierung. Die Boxrunden werden von einer knapp bekleideten Dame eingeläutet, dann schreiten die Boxer zum Kampf. Vier Grundschläge stehen zur Auswahl.

RINGSIDE ist keine reine Boxsimulation, strategische Elemente bereichern das Spiel. Die eigentliche Simulation läßt etwas zu wünschen übrig, zu träge bewegen sich die Boxer durch den zweidimensionalen Ring. Schnelles Ausweichen, um feindliche Treffer zu vermeiden, ist fast unmöglich. Boxhandschuhe geben Aufschluß über die Bewertung der Richter. Eine Skala zeigt die körperliche Verfassung. In den Pausen kann der Boxer jedoch neue Energie erhalten. Gegenüber den spielerischen Mängeln besitzt RINGSIDE eine ausgezeichnete Grafik. Auch der Sound ist als gut zu bezeichnen, beispielsweise klatschen die Zuschauer gegen Ende einer Runde rhythmisch in die Hände. Überzeugen konnte RINGSIDE trotz allem nicht, lange Ladezeiten auch während des Spiels und die schon erwähnten spielerischen Mängel sind dafür verantwortlich. Wer Boxspiele liebt, kann sich das Spiel trotzdem einmal näher anschauen und sich eine eigene Meinung bilden.



Die beiden Boxer bewegen sich etwas träge durch den Ring.



Anbieter: EAS

'Sherlock - The Riddle of the Crown Jewels' ist das erste Adventure einer Reihe, die unter dem Motto 'Immortal Legends' steht. Sie soll die Abenteuer berühmt gewordener Persönlichkeiten in Adventure umsetzen. Infocom reagiert damit auf eine Umfrage unter den Lesern der firmeneigenen Clubzeitschrift 'The Status Line', die früher 'The New Zork Times' hieß.

Sherlock

```
 Entry Hall                          Saturday   5:01:00 a.m.       Score:     0
Hudson. As the fog swirls around you, you huddle into your coat and shiver in
the predawn chill.

>go west
The door is not open.

>knock
[What do you want to knock at?]

>knock door
You hear a bolt sliding back and the door opens to reveal a worried Mrs Hudson.

"Oh Doctor Watson, I'm so glad to see you. It's Mr Holmes, sir. Three days have
come and gone since he's been out of his rooms. He won't eat and I know he
hasn't slept. Now there's a visitor who says he's here about some important
government business. I tried to send him away, but he insisted on waiting in
the parlour. That's when I sent for you. Thank God you've come."

You step inside and Mrs Hudson closes the door behind you. The entry hall is
just as you remember it - small, underheated, and lit by a feeble gas light.
The door to the parlour is to the north, and a flight of well-worn stairs leads
up to the first floor.

>█
```

Die Hintergrundstory sieht etwa wie folgt aus: Ganz England befindet sich in den Vorbereitungen zum 50. Jubiläum der Queen Victoria. Es versteht sich von selbst, daß ihre Majestät bei dieser Gelegenheit die Kronjuwelen tragen wird. Dummerweise hat sich jemand ausgerechnet diesen Zeitpunkt ausgesucht, die Kronjuwelen aus dem Tower von London zu klauen. Dann hat sich dieser Frechling auch noch erlaubt, am Ort des Verbrechens ein Gedicht zu hinterlassen, in dem ausdrücklich auf Sherlock Holmes hingewiesen wird.

Aus diesem Grund wendet sich der Premierminister an den großen Detectiv. Er übernimmt natürlich den Fall. Allerdings erscheint es ihm aufgrund des Gedichtes wahrscheinlich, daß der Dieb ihn genau kennt, darauf spekuliert, daß er die Verfolgung aufnehmen wird und ihm dementsprechende Fallen gestellt haben muß. Darum übergibt er Dr. Watson, also dem Spieler, die Leitung der Ermittlungen. Dieser, als vollkommen andere Persönlichkeit, wird ganz anders vorgehen als Sherlock Holmes und somit alle Fallen und Sackgassen umgehen. Eine Schwierigkeit ist noch die Tatsache, daß Sie nur zwei Tage, sprich 48 Stunden, Zeit haben, die Kronjuwelen zu finden. Denn dann beginnen die oben erwähnten Festlichkeiten, und sollte es der Queen nicht möglich sein, ihre Juwelen zu tragen, würde die Regierung und somit das gesamte Empire international in Mißkredit fallen.

Sie haben jetzt die Möglichkeit, alle denkbaren Schauplätze in London aufzusuchen und Ihre Ermittlungen durchzuführen, angefangen beim Tower über Madame Tussauds Wachsfigurenkabinett bis hin zu Westminster Abbey, Scottland Yard und dem Diogenes Club. Sie können sich mit der Kutsche oder auch zu Fuß durch das nebelige, nächtliche London bewegen. Tagsüber werden Sie von Touristen behindert, auf dem Trafalgar Square wimmelt es von Tauben. Man fühlt sich regelrecht in Sherlocks Zeiten versetzt.

Um sich so richtig ins London der damaligen Zeit einzustimmen, liegt der Packung auch noch eine Londoner Zeitung aus dieser Zeit bei, mit aktuellen Artikeln und Werbung, außerdem eine Karte von London und ein toller Schlüsselanhänger.

Nach einem on-screen-mapping bei Beyond Zork und nach Adventures, die aus mehreren Einzelgeschichten bestehen, erinnert Sherlock wieder an die früheren Infocom-Adventures. Allerdings enthält Sherlock wie auch schon andere die Hints-Funktion. Für mich ist das der größte Nachteil des Spiels. Denn nur, wenn man wochenlang an einem Problem nagt und selbst die verrücktesten Ideen ausprobiert, findet man die meisten der von den Programmierern eingebauten Gags. Und die machen das Spiel spielenswert.

Hat man dagegen die Möglichkeit, für wirklich jedes Problem die Lösung aus einer Bibliothek von Lösungen auszuwählen, beschränkt sich das Lösen des Adventures auf ein bloßes Durchspielen. Normalerweise beschäftigt ein Adventure den Spieler für mehrere Wochen. Unter Zuhilfename der Hints kann man Sherlock innerhalb eines Nachmittages durchspielen. Es gibt zwar die Möglichkeit, die Hints-funktion abzuschalten, aber wer wird an einem Problem eine Woche lang knobeln, wenn er innerhalb weniger Sekunden die Lösung auf den Schirm bekommen kann?

Abgesehen davon ist Sherlock nicht das schlechteste Infocom-Adventure.

In der getesteten MS-DOS-Version gab es keinen Sound, aber laut Infocom sollen die Versionen von ATARI ST und AMIGA mit Soundeffekten erhältlich sein.

Andreas Paul



INSERENTEN-VERZEICHNIS

KICKSTART PU

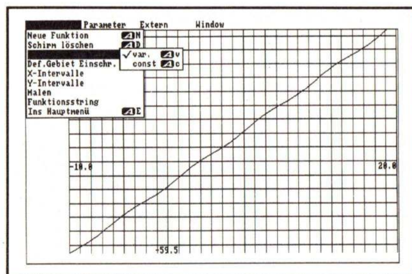
Hallo PD-Fans,

mit Volldampf gehen wir ins neue Jahr und es soll noch einiges passieren. Wir wollen vermehrt Berichte über gute und wichtige PD-Programme bringen. Wenn Sie über solche Programme berichten können weil Sie es schon länger benutzen, dann setzen Sie sich mit mir in Verbindung. Ansonsten finden Sie auf den neuen 10 KICK-PDs wieder ein breites Angebot an Programmen und Utilities.

Bis zum nächsten Mal
Markus Nerding

DIE NEUHEITEN

KICK 121: MODULA (AMOK 11)



R.o.M.: ein komfortables Mathematikprogramm und Funktionsplotter
TrackDiskSupport: erleichtert Programmierung der TrackDisk
M2Druka: Druckutility für ASCII-Dateien
Beep: erzeugt akustisches Warnsignal

KICK 122: MODULA (AMOK 12)

Auch auf dieser AMOK-Diskette gibt es wieder jede Menge Routinen und Programme: Modul zum Ansteuern des Fischertechnik-Interfaces, ein schnelles und sicherers HD-Backup, Disassembler für M2Amiga-Object-Files, Array-Sort, Janus, Timer, StripText, ConvLibSym, MakeInLine, Diff, Tips zum M2-Compiler

KICK 123: UTILITIES

Neue Utilities, die aus den aktuellen Einsendungen zusammengestellt werden. Hier gibt es immer ein riesiges Angebot, denn der AMIGA läßt viele Möglichkeiten für Hilfsprogramme.

KICK 124: ANWENDUNG



STARCHART: umfangreiches und interessantes Sternenprogramm mit unendlich vielen Einstellungen. Mit der Maus können Informationen über die Sterne abgerufen werden.
AIRFOIL: ein einfacher Windkanalsimulator, verschiedene Einstellungen.

KICK 125: GRAFIK



SLIDESHOW: Ein komfortables Diashow-Programm für DPaint-Bilder (320x200). Alle Bilder des aktuellen Direktories werden nacheinander auf Mausklick angezeigt, wahlweise auch rückwärts. **DARK**: Copper-Demo, die viele Elemente eines Ballerspiels enthält. C-Source. **GLOBE**: Eine kleine drehende Weltkugel. Pop-Up-Menu, Source in C. **ICONS**: verschiedenste, animierte Icons (Disk, Drawer, Tool, Project, Garbage). Sehr originell und lustig!

KICK 126: GRAFIK



Mandelbrot-Show I: eine sehr informative Reise durch die Welt der Mandelbrotgrafiken. Jedes Bild wird ausführlich erläutert und das jeweils darauffolgende ist eine Vergrößerung eines Ausschnitts davon. Mit Mandelbrotgenerator MSE. Autor: R. Beck

KICK 127: GRAFIK

Mandelbrot-Show II: dies ist der zweite Teil der Reise durch die Welt der Mandelbrotgrafiken. Autor: R. Beck

KICK 128: SOUNDS

Scores, Drumkits und Soundfiles für Dynamic Drums von New Wave Software.

KICK 129: SOUNDS



Neue Kompositionen von Andreas Starr aus Gronau, erstellt mit Sonix. Effektvolle Stereostücke mit Player.

KICK 130: ANWENDUNGEN



SUPERLIGA: eine Fußballergebnisverwaltung mit 577 Mannschaften aus 23 Ländern und 34 Ligen. Über 10000 Ergebnisse können gespeichert werden. Die Eingabe gestaltet sich sehr einfach. Aus den bisherigen Ergebnissen kann eine Prognose (Regressinsanalyse) für kommende Spiel und die Meisterschaft erstellt werden. Autor: Rolf Morlock

Machen Sie mit!

Möchten auch Sie selbstgeschriebene Programme der Allgemeinheit zur Verfügung stellen, so schicken Sie sie uns einfach zu (bitte mit Dokumentation auf Diskette!).

Als kleine Anerkennung können Sie sich dafür 5 Disketten aus unserem PD-Service auswählen.

MAXON-Computer GmbH
KICKSTART-Redaktion
PD-Einsendung
Industriestraße 26
6236 Eschborn

BLIC DOMAIN

KICK 120: UTILITIES
PCopy, HexCalc, Dux5 (CLI-Utility), SDBackup, FlamKey (Computerschloß), VirusX, SCT (Farbeinstellung von WB), Surveyor, HPMan (totale Kontrolle über den Laser)

KICK 119: AMOK #10
FlightDemo u. 3D-Demo (Echtzeitdemonstrationen bewegter räumlicher Grafik - super), FileRequester, SuperLists1.3, Speech (dt. Sprachausgabe), Feigenbaum.

KICK 118: AMOK #9
GraphicSupport (Makros für Graphics-Primitives/Intuition), PrintIt2.0 (Hardcopyroutine für Epson), M2Test (Dokumentation über Fehler bei ARRAYs mit Index BOOLEAN), TextFont (Laden v. Zeichensätzen), Sounds und Graphics.

KICK 117: AMOK #8
MemSystem (Speicherverwaltung), PrinterSupport (unterstützt Printer.device), IDCMP (Hilfsroutinen), Blitter (direkte Blitterprogrammierung), SpaceDemo (3-D Demo), IFF8SVXLoad (Laden/Abspielen gesampelter Sounds).

KICK 116: AMOK #7
Reminder (Wecker für Programmierer), WarpText (schnelle Textausgaberoutine), MuchMore (Softsroll), M2Test (Hinweise auf Fehler in den Bibliotheksmodulen), SoundTask, AVL-Bäume, Listen, Queues, Stack.

KICK 115: SPIELE
CHEES-BOXES ist das allseits bekannte "Käsekästchen"-Spiel, das man jedoch nicht mit dem folgenden verwechseln sollte. 2-4 Spieler, mausgesteuert, AMIGABasic. Autor: J.D. Mallander. **VIER GEWINNT**: zwei Spieler, mausgesteuert, AMIGABasic. Autor: Björn Kessels. **COSMIC**: Strategiespiel. Source in C. Autor: Carl Edman. **SOLITAIRE** löst beliebige Stellungen des bekannten Spiels. Source in C. Autor: Thomas Lagally **GRID IT**: spannendes Rennen für zwei Spieler. Benötigt Joystick, Source in C. Autor: Jörg Tuttas **LABYRINTH II**: Textadventure in Stil der INFOCOM-Spiele. Englisch, C.

KICK 114: BILDER-SHOW
Eine Grafik-Show mit Mandelbrotbildern und Colorcycling. Dadurch wird die Wirkung der eindrucksvollen Bilder hervorgehoben. Autor: Hermann Dörries, Wildeshausen

KICK 113: SOUNDS
Auf dieser Diskette ist der zweite Teil der KICKSTART SOUND DEMO III mit den Stereo-MIXES: Return to Defender, Dark Castle Mix, Chicago-Tune, Disco-Mix und Crazy Tune. Wichtig: Die Diskette läuft nur mit KICK 112!

KICK 112: SOUNDS
Zum dritten Mal haben unsere Soundspezialisten TOB & TWP zugeschlagen. Das Ergebnis ist die KICKSTART SOUND DEMO III mit einer Menge hervorragender Sounds auf zwei Disketten (siehe KICK 113). Mit von der Partie sind in der Rubrik CHARTS: Relax, Jean Michelle Jarre, Every Breath You Take, Axel F., Push, Tell It To My Heart und Blue Monday. Alles HiFi-Stereo mit einer extralangen Spielzeit.

KICK 111: BILDER
20 wunderschöne IFF-Grafiken von Frauenprofilen. Mit Diashow, sehenswert!

PD 110: UTILITIES
MEMORYCLOCK: zeigt Uhrzeit, Datum, RAM- und Chip-Memory. **DATAMAKER, IFF-CONVERTER**: wandelt Dump-Format-Bilder in IFF-Format. Source zu allen Programmen in Assembler, Autor: Roger Fischlin.
SDBACKUP: Festplatten-Backupprogramm, das die komplette Platte in komprimierter Form auf Disketten abspeichert. Viele Optionen. **RUNBACK:** läßt Programme im Hintergrund laufen, damit sie keine CLI-Fenster benötigen.

PD 109: SCHULE
VOK-BOY V1.0: Vokabeltrainer in AmigaBasic. Autor: Michael Hennemann
WORD: Fremdsprachenlernprogramm, das Begriffe und ganze Sätze abfragt. Mit umfangreichen Lektionen in Englisch und einigen in Französisch. Kann leicht erweitert werden. Verwaltet beliebige Sprachen, sehr komfortabel. Autor: Daniel Neukomm, Bern (Schweiz)

PD 108: BILDER
Viele schöne Bilder zum Anschauen und Weiterbearbeiten.

PD 107: ANIMATIONEN
Weitere Animationen von Tobias Richter: JET (Düsenjet-Landeanflug), RELIANT (ein Raumschiff im frontalen Anflug), NELSON (rotierendes Raumschiff). Die Animationen demonstrieren, wie eindrucksvoll die mit VideoScape erstellten Sequenzen sein können. Sehenswert!

PD 106: ANIMATIONEN
Tobias Richter hat uns wieder einige seiner mit VideoScape 3D-Animationen geschickt: LOTUS (der bekannte Lotus in einer neuen Sequenz), WILKERSON, DURETT und RELIANT (Raumschiffe in schneller Rotation).

PD 105: ANIMATIONEN
LLEWELLYN ist eine mit dem Sculpt 3D-Animator erstellte Filmsequenz mit musikalischer Untermalung. Eine schöne Demo dieses Programms von Michael Clinard.

PD 104: ANWENDUNGEN
DIRECTORY MASTER V1.1: Dieses Programm bringt Ordnung in Ihre Diskettensammlung. Die Dateien jeder eingelegten Diskette werden automatisch in eine Liste übernommen. Diese kann dann auf komfortable und vielfältige Weise bearbeitet und verändert werden. Sehr viele Funktionen, einfache Bedienung, sehr leistungsfähig und schnell. Als Zugabe gibt es die jeweils aktuelle Liste des KICKSTART PUBLIC DOMAIN SERVICE.
MEMOPAD: ist ein echt nützliches Hilfsprogramm für vergeßliche AMIGA-Anwender. Hier können wichtige Termine und Daten festgehalten werden. Bindet man den Befehl CHECKMEMO in die Startup-Sequence ein, dann erinnert das Programm an diese Termine. Einfache Mausbedienung, sehr komfortabel und hilfreich.

PD 103: GRAFIK
C-LIGHT: ein sehr einfach zu bedienendes, mausunterstütztes Raytracer-Programm. Die Einstellungen erfolgen über viele Regler und Schalter. **NEWJUGGLER**: ein Grafikprogramm, das den Eric Graham's JUGGLER innerhalb eines Bildes ablaufen läßt. Source in C; Autor: Peter Weiland, Wietze. **HAM'S**: einfaches Malprogramm für den HAM-Modus in AmigaBasic. Das Programm kann leicht geändert werden, da der Source-Code gut strukturiert ist. Autor: Henning Frommer, Vallendar

PD 102: SPRACHEN
STONY BROOK PROLOG: auf dieser Diskette befinden sich die Source-Codes (Lattice C 4.0 und Prolog) des PD-Prolog-Systems von Diskette 101, mit denen man Veränderungen und Erweiterungen vornehmen kann.

PD 101: SPRACHEN
STONY BROOK PROLOG Version 2.3.2: Dieses PD-Prolog-System basiert auf einem System, das ursprünglich für UNIX-Rechner konzipiert wurde. Das System beinhaltet sowohl einen Interpreter als auch einen Compiler (bis zu 3000 LIPS). Der Sprachumfang entspricht weitgehend dem Edinburgh-Standard. Das System besitzt zusätzlich einige Besonderheiten z.B. dynamische Prädikateinbindung, Mischen von interpretiertem und compiliertem Kode, Makros und andere Hilfen. Inklusive ca. 100seitiger Dokumentation. Benötigt mindestens 1MB Speicher.

PD 100: JUBEL-DISK
Auf dieser Diskette sind Programme zusammengefaßt, von den wir denken, daß sie für einen AMIGA-Besitzer unverzichtbar sind. Die Palette reicht von Kopierprogrammen, CLI-Hilfen bis zu Spielen. Lassen Sie sich überraschen - es wird sich lohnen!

PD 99: SPIELE
BrainWork: Puzzle. *Autor: Timmy.* **Amiga-Wurm**: Tron-Variante; AmigaBasic. *Autor: Christoph.* **May Valley of the Aztec's**: ähnlich "Kaiser"; ABasic. *Autor: Matthias Hakuba* **Bauernskat**: Source in AmigaBasic. *Autor: Oliver Peter* **Breakout**: 99 Level mit Editor; Source in C. *Autor: Peter Händel.* **Kniffel**: bekanntes Würfelsspiel für 4 Personen oder gegen Computer; sehr schöne Grafik. Mausbedienung. *Autor: Michael Teistler.* **PACCIE**: Pacman-Variante in Basic.

PD 98: SUPER-SOUNDS
KICKSTART SOUND DEMO II: sicherlich eine der besten Sounddemos, die es derzeit auf Disk gibt. 18 programmierte Musikstücke, aktiongeladener Sound, Stereo, sehr lange Spielzeit, kurze Ladezeit durch gepacktes Format, Sound-Slideshow. *Komponist: Tom Beuke*

PD 97: UTILITY
TASKMON V0.4: zeigt die Zustände aller Tasks; Source in C. *Autor: P. Erpenbeck.* **FilterSwitch 2.0**: damit läßt sich der Tiefpaßfilter per Alt-Taste an- und ausschalten. **DMASwitch 2.0**: Bildschirmabschalter; Source in C. *Autor: Andreas Jung.* **MouseTool** (Fenster hervorholen, vergrößern, verkleinern, PALPatch (verändert Intui.-Lib., vergrößert alle Screens und Windows auf PAL-Größe), ChangeTaskPri, CLI4WB (CLI-Programme von der WB starten); alle Prg. mit Source (Profimat-Assembler). *Autor: Richard Englert, Fürth.* **RossiDress V1.0**: benutzerfreundliche Adressenverwaltung; AMIGABasic. *Autor: Jürgen Rogg.* **Milben**: kleiner Screenhack; Source in C. *Autor: Guido Appenzeller*

PD 96: UTILITY
TurboBackup: komfortables und schnelles Kopierprogramm; kopiert auf bis zu 3 Laufwerke mit Verify.
MRBackup2.1: Harddisk-Backupprogramm, sichert auch überlange Dateien.
DMouse: Bildschirmschoner, Mausabschalter, Mausbeschleuniger, Fensteraktivator, Funktionstastenbelegung.
ARC: Programm zum Packen und Entpacken von Dateien. Hiermit können Programme platzsparend abgelegt werden (siehe PD 94/95).

PD 95: ANWENDUNGEN
AnalytiCALC: professionelles Tabellenkalkulationsprogramm (Spreadsheet), das auch schon für MS-DOS, VAX und PDP-11 existiert. Das Programm arbeitet im Interlace-Modus und benötigt deshalb 1MB Speicher. Sehr leistungsfähig (18.000 * 18.000 Zellen usw.).
Wichtig: Die Dateien der Disketten 94 & 95 sind in komprimierter Form abgelegt. Sie benötigen das Programm ARC (PD 96), um sie wieder zu entpacken.

PD 94: ANWENDUNGEN
RIM-5 ist ein relationales Datenbanksystem, für Verwaltung großer Datenmengen. Integrierte Programmiersprache. Help-Funktion. Incl. 80-seitiger Anleitung und Source in Absoft-Fortran .

PD 93: MODULA-2 (AMOK #6)
SoftScroll (zum Scrollen einer BitMap), MathLib (Bibliothek für Vektoren und Matrizen), IFFSupport (Laden und Speichern von ILBM-Bildern), ListAll (erweitert LIST für Subdirectories)

PD 92: MODULA-2 (AMOK #5)
Graphics (IFF-Bilder mit Mausklick), PrinterSupport, Print (TYPE TO PRT:), MemSystem (intelligenter Heap-Ersatz), Superlist1.2 (für Scrollbar-Requester), PrintIt! (Hardcopyprogramm), IconSupport (zum Erzeugen von Icons)

PD 91: MODULA-2 (AMOK #4)
CHECKER ist im Rahmen einer Studienarbeit im Fach Technisches Design entstanden. Thema war die "Erarbeitung neuer Möglichkeiten der Informationsdarbietung in Kraftfahrzeugen unter Zuhilfenahme eines microcomputergestützten Grafikbildschirms".

PD 90: UTILITIES
PERFECTSOUND V2.1: Ein umfangreicher Soundsampler mit vielen Features (z.B. Stereo samplen, Stereo erzeugen, Instrument erzeugen, Bereiche manipulieren, umkehren, löschen uvm.). Einfache Mausbedienung (siehe auch Bericht in KICKSTART 10.88).
CROSSREF: Dieses Programm erzeugt eine Aufstellung aller Variablen und Befehle mit Zeilenverweis. Unverzichtbar für Programmierer; verwendbar für alle ASCII-Dateien. M2-Modula. *Autor: Jörg Gutzke, Mönchengladbach*
DISKLAB: ein Programm zum Drucken von Disketten-Labels auf 9 Nadel-Druckern (Anpassung an andere möglich!). Mausgesteuerte Auswahl von Direktoryeinträgen, Grafikeinbindung, sehr komfortabel. Source in AMIGABasic. *Autor: Manfred Illi, Ludwigshafen.*
BOOTBLOCK-CHECKER: automatisches Testen des Bootblocks, akustischer Alarm, ASCII-Ausgabe u.a. *Autor: Matthias Kühn, Weinheim*

PD 89: DIGI-SOUNDS
The Sound of Music (Volume 3): weitere digitalisierte Musikstücke von *Harald Schneider,* mit Player. Hitparadensounds für eigene Programme und Vorspänne.

PD 88: MUSIK
Weitere Musikstücke von *Andreas Starr,* die man nicht verpassen sollte. Stereo, mit Player.

PD 87: MUSIK
Sehr schöne Musikkompositionen schickte uns *Andreas Starr* aus Gronau. Er hat sie mit Sonix erstellt. 5 Stücke (u.a. Miami Vice-Variation) in echtem Stereo mit einer Spielzeit von fast 20 Minuten. Die Musik wird im Hintergrund abgespielt. Lassen Sie sich dieses Erlebnis nicht entgehen.

PD 86: SPIELE
CYCLES 1.0: Eine sehr spannende Tron-Variante, bei der man sein Fahrzeug um Gegner und Quadrate herumlenken muß. In höheren Leveln wird die Geschwindigkeit größer, und immer mehr Quadrate erschweren die Suche nach einem freien Weg.
ESCAPE FROM JOVI: Wer wagt mit dem Raumschiff die Flucht durch die engen Gänge von Jovi? Source in Assembler. *Autor: Oliver Wagner, Sprockhövel*
ZAUBERFLÄCHEN: Erfordert gute Taktik, die verschiedenen Farbfelder zu sortieren. Kein Spiel für Leute, die gleich verzweifeln. Source in AMIGABasic. *Autor: Heiko Jahn*

PD 85: SPIELE

Eines der besten und spannendsten Strategiespiele ist **RISK V3.0**. Bis zu 5 Spieler müssen versuchen, die Herrschaft über die Welt zu erlangen. Dabei entscheidet nicht nur das Glück, sondern es kommt auch auf geschicktes Verhandeln an. Besonders wenn viele mitspielen, ist es schnell um einen geschehen, bzw. man wird nie sein Ziel erreichen. Das Spiel hat eine sehr schöne Grafik, ist sehr einfach mit der Maus zu bedienen und hat eine Vielzahl von Optionen, die den Spielablauf erleichtern oder erschweren können. Ein ungeheurer Spaß für eine lustige Spielerrunde. *Autoren: Michael Berling & Hartmut Stein (Bernstein Zirkel Softworks)*

PD 84: BILDER

Eine Bildershow mit fantastisch digitalisierten Bildern, Sound und Effekten. Die sehr realistischen und kunstvollen Aufnahmen zeigen was man, mit der entsprechenden Ausstattung, auf dem AMIGA machen kann. Erstellt mit dem DVS-2000 System von PBC Computerdesign.

PD 83: MODULA-2 (AMOK #3)

IFF-Loader in Assembler, Konverter für IFF-Brushes, Rekorder für Maus- und Tastenaktionen, Tool für GELs und anderes mehr.

PD 82: MODULA-2 (AMOK #2)

Die zweite Diskette des Modula-Klubs AMOK mit Routinen zu folgenden Themen: IFF-Loader, Halfbrite- und HAM-Demo, Hilfe zur Intuitionprogrammierung, Requesterroutinen.

PD 81: MODULA-2 (AMOK #1)

7 begeisterte Modula-Fans aus dem Raum Stuttgart haben sich zum Amiga Modula Klub (AMOK) zusammengeschlossen. Ihre Erfahrungen machen sie jedem Interessierten über die AMOK-Disketten zugänglich, auf denen sich hilfreiche Routinen in Form von Modulen befinden. Zu jedem Modul gibt es eine ausführliche Anleitung und ein Beispielprogramm. Auf der ersten Diskette befinden sich Programme wie z.B. Sprite- und Bob-Handhabung, Joystick-Routine, Interruptsteuerung und vieles mehr.

PD 80: UTILITIES

BOOTBLOCK CHAMPION testet den Bootblock auf Viren und erlaubt es, Assemblerprogramme darauf zu schreiben. Außerdem kann der Bootblock gesichert werden. *Autor: Roger Fischlin*

VIRUSTEST und **VIRUSCHECK** sind Virustestprogramme, die in die Startup-Sequence eingebunden werden können. *Autor: Roger Fischlin*

DFC ist ein einfach zu bedienendes, aber leistungsfähiges Kopierprogramm. Formatieren, Verify und Kopieren auf bis zu 4 Disketten sind die Optionen des Programms.

PD 79: SCHULE

PROCALC 1.1 ist eine komfortable Taschenrechnersimulation eines HP11G. Die Bedienung solcher Rechner ist einfach, jedoch anfangs gewöhnungsbedürftig (UPN). Der große Leistungsumfang wird in einer ausführlichen deutschen Anleitung erklärt. *Autor: Götz Müller*

GRAPH-MASTER V3: ein sehr umfangreiches Programm zum Darstellen von Funktionen. *Autor: Rolf Beck*

3-D FUNKTIONEN ist ein komfortabler und schneller 3D-Plotter. Source in AMIGABasic.

PD 78: GRAFIK-SHOW

FRAKTAL-MOVIE ist ein Film aus 200 s/w-Fraktalgrafiken, die hintereinander gezeigt werden (ca. 8 Bilder/s). Dabei wird aus einem Fraktal langsam herausgezoomt, bis man das nächste erkennen kann usw.. Benötigt für den kompletten Film 1MB Speicher! *Autor: Jürgen Weineit*

PD 77: SPIELE

KART ist ein Autorennspiel, bei dem zwei Spieler ihren jeweiligen Wagen über 6 verschiedene Rennstrecken lenken müssen. Höchst-, Lenkgeschwindigkeit und die Rundenanzahl können vorgewählt werden. Gute Grafik und Sounds; erfordert zwei Joysticks. Source in C. *Autor: Axel Ilenburg*
BUNDESLIGA ist ein Spiel, bei dem eine Fußballmannschaft zur Meisterschaft geführt werden soll. Geschicktes Kaufen und Verkaufen und die Aufstellung des Teams sind die wichtigsten Maßnahmen für den Erfolg. Source in AMIGABasic. *Autor: Ingo Hirsch*

PD 76: SPIELE

KAMPF UM ERIADOR V2.0 ist ein Fantasy-Strategiespiel für zwei Personen. Im Jahre 6735 wogt der Kampf zwischen Gut und Böse im Land Eglador. Auf einem großen Feld stehen sich die Legionen der beiden Seiten gegenüber. Mit strategischem Geschick müssen Sie versuchen, den gegnerischen Magier zu erreichen, der sich am anderen Ende des Spielfeldes in seiner Festung befindet. Gute Grafik und Sounds; benötigt 1MB Speicher! *Autor: Ralf Böwing, Roland Hartz*

PD 75: SUPER-SOUNDS

KICKSTART SOUND DEMO: Nicht digitalisierte Sounds, sondern programmierte Musikstücke schickt uns Tom Beuke, damit wir "Abstand nehmen von den schnöden Digi-Sounds (kann doch jeder, oder?)". 17 actiongeladene Musikstücke; Länge 12-95kB. *Komponisten: TOB & TWP*

PD 74: DIGI-SOUNDS

The Sound of Musik (Volume 1): Digitalisiert Musikstücke zur Verwendung in Vorspännen, Demos und eigenen Programmen. Sehr gute Qualtität; mit Player; 11 Samples; Länge 30-100kB. *Sampler: Harald Schneider*

PD 73: SPIELE

LIGHT CYCLE 3D: ist eine sehr gut gemachte TRON-Variante, bei der zwei Fahrer versuchen, dem Gegenspieler den Weg abzuschneiden. Die 3D-Grafik ist dabei ein wirkliches Erlebnis. Da jeder Fahrer eine undurchdringbare Bahn hinterläßt wird das Spielfeld schnell eng. Mit vielen Anzeigeinstrumenten und Anleitung. *Autor: Tobias Richter*
ARTILLERIE-DUELL: in einer zerklüfteten Berglandschaft sehen sich zwei Kanonen gegenüber. Jeder der zwei Spieler muß nun den Abschußwinkel und die Stärke einstellen, mit der er seine Kugel abfeuern will. *Autor: Peter Kliem*
TREFFER: eine Kniffel-Variante, die komplett mit der Maus gesteuert wird. Bis zu vier Spieler, mit Zwischenstand, Hi-Score und Statistik. *Autor: Michael Jänecke*

PD 72: ANIMATIONEN

Für alle **STAR TREK**-Fans hat Tobias Richter zwei Animationen mit VideoScape 3D erstellt. Die erste zeigt ein Klingonenschiff, die zweite die U.S.S. ENTERPRISE.

PD 71: UTILITIES

PRTDRVGEN 2: der Druckertreibergenerator zum Anpassen der exotischsten Typen an den AMIGA. Sehr einfache Benutzerführung mit ausführlicher (englischer) Erläuterung aller Optionen (siehe Bericht in KICKSTART 8/9 88).
VIRUS X: läuft im Hintergrund und prüft den Bootsektor jeder eingelegten Diskette auf Viren und mögliche Veränderungen. Mit Source.
Overscan: ein Patch für die Intuition-Library, mit der amerikanische Programme das PAL-Format ausnutzen könnten, ohne daß sie modifiziert werden müssen.
HP: ein RPN-Taschenrechner mit vielen Funktionen (BIN, OCT, DEC, HEX, SIN, COS, TAN, ATN, LOG, LN, LG2, ATN, y^x ...) und 32 Registern. Mit Source in C

PD 70: SPIELE

WHEEL: Bei diesem Spiel muß ein Ausdruck erraten werden. Diese Glücksrad-Simulation ist grafisch gut aufgemacht und komplett mausgesteuert. Da die zu erratenden Begriffe in englisch sind, ist dieses Spiel gut zum Erlernen dieser Sprache geeignet.
MASTERMIND: Kombinationsspiel, bei dem eine Farbkombination erraten werden muß. Mausgesteuert!
RISTINOLLA: Eine spielstarke GO-Muko-Variante. Mausgesteuert und sehr schnell!

CLUE: das bekannte Brettspiel für pfiffige Detektive in einer ansprechenden Computerversion. Verschiedene Räume müssen untersucht, Personen befragt und die Ereignisse kombiniert werden um den Mörder herauszufinden. Die gute Grafik und die einfache Mausbedienung machen eine Anleitung weitgehend unnötig.

PD 69: GRAFIK-DEMOS

AMUC_DEMO: Ein Bild mit 200 x 2000 Bildpunkten scrollt vertikal über den Bildschirm. Ein sehenswertes Erlebnis.
HBHILL: Dies ist Kevin Sullivans Beitrag zum BADGE KILLER DEMO CONTEST. Die Demo nutzt den Extra-Half-Brite-Modus (64 Farben) des AMIGA.

JETZT NUR NOCH
DM 8.-
PRO DISKETTE

Wir verwenden nur doppelseitige Markendisketten der Firmen JVC und MAXELL.

HAMmmm: Eine mit Musik unterlegte Linendemo von Phil Burk. Inclusive Source-Code in JForth. **STARS**: Ein interessanter Flug durch einen Sternenhimmel. **WIREDEMO**: zeigt einen rotierenden Linienkörper, mit Source in C.

PD 68: GRAFIK-DEMOS

WBLANDER: versuchen Sie das Raumschiff sicher auf einem Workbench-Fenster zu landen. Auch dieses Programm hat am BADGE KILLER DEMO CONTEST teilgenommen
DRUNKENMOUSE: wenn Sie nicht betrunken sind, dann liegt es sicherlich an diesem Programm, daß Sie Schwierigkeiten haben, etwas anzuklicken.
MACHINE: eine hervorragende Demo von Allen Hastings-erstellt mit VidoeScape 3D.

PD 67: UTILITIES

PR: ein Programm zum Ausdrucken von Texten. Das Inhaltverzeichnis wird in einem Fenster dargestellt und das File kann mit der Maus ausgewählt werden. Auf dem Ausdruck erscheint dann eine fortlaufende Seitennummerierung, der Name und das Datum.
SHOW: entspricht dem Programm PR jedoch werden die Texte nur angezeigt.
FUEL GAUGE: ist eine grafische Anzeige für den vorhandenen und benutzten Speicherbereich.
SIMCPM: CP/M-Emulator, der den 8080-Prozessor emuliert. Mit H19 Terminal.
LOGO: ein Interpreter der bekannten Turtle-Spache, die mehr kann als nur malen.
MICROSPELL: ist ein flexibler Spellchecker mit einem englischen Grundwortschatz von 43.000 Wörtern.

PD 66: M2-Modula

M2 AMIGA: Dies ist die funktionsfähige Demo-Version des M2-Modula-Compilers. Beschränkungen gibt es nur hinsichtlich der Größe der compilierbaren Programme, außerdem sind nur wenige Schnittstellen und Standard-Bibliotheken vorhanden. Auf der Diskette sind einige Demo-Programmme enthalten, die z.T. direkt für diesen Compiler entwickelt wurden.
Der MODULA-Kurs, der in dieserer Ausgabe der KICKSTART beginnen wird, kann fast vollständig mit diesem Compiler nachvollzogen werden.

PD 65: ICON-TOOLS

IDPICON: ermöglicht es, eine mit Deluxe Paint erstellte Brush beliebiger Größe in ein Icon umzuwandeln.
SNAP: wandelt ein beliebiger Bildschirm in ein Icon um. Die Größe des Icons ist dabei einstellbar. Bei mehrfarbigen Screens wird eine Farbreduzierung durchgeführt.
ALTICON: gibt dem eigenen Icon den letzten Pfiff, denn hiermit werden zwei Icon-Bilder zu einem 'animierten' Icon zusammengefügt. Außerdem viele schöne Icons.

PD 64: (Strategie-) SPIELE

BACKGAMMON: bei diesem Strategiespiel können Sie zeigen, ob Sie besser sind als der Computer.
PUSH OVER: hier wird so lange geschoben bis plötzlich und unerwartet einer der Spieler fünf Steine in einer Reihe hat. Ein spannendes Strategiespiel.
PUZZLE PRO, IFF2PICS: zwei Programme, die ein beliebiges IFF-Bild in viele Einzelteile zerlegen. Dann liegt es an Ihnen, wie lange Sie brauchen, um es wieder zusammenzupuzzeln. Einstellbarer Schwierigkeitsgrad.

PD 63: (Action-) SPIELE

AMOEBA: hinter diesem Namen verbirgt sich eine spielhallenmäßige SPACE INVADERS-Adaption. Obwohl dies ein Spiel der ersten Computergeneration ist, hat es nichts von seiner Spannung verloren. Ein sehr gutes Ballerspiel.
ASTEROIDS ist ein Weltraumspiel, das ebenfalls von Spielhallencomputern umgesetzt wurde. Es ist nicht ganz einfach, die Kontrolle über das Raumschiff zu behalten und es sicher im Asteroidenhaufen zu manövrieren.
EGYPTIAN RUN: Actionspiel, bei dem ein Geländewagen durch die Wüste gesteuert wird. Dabei muß man den Sandhügeln und besonders dem Sandstrahl ausweichen, der den Wagen zu unkontrollierten Bewegungen veranlaßt.

VERSANDBEDINGUNGEN:

Um einen schnellen und problemlosen Versand zu gewährleisten, beachten Sie bitte folgende Punkte:

- Bestellungen per Nachnahme oder Vorrauskasse
- Für jede Diskette ergibt sich ein Unkostenbeitrag von DM 8.-
- Pro Sendung kommt ein Versandkostenbetrag (für Porto und Verpackung) von DM 5.- (Ausland DM 10.-) hinzu.

Bei einer Bestellung von 5 oder mehr Disketten entfällt der Versandkostenbeitrag

- Bei Nachnahme zuzüglich 3.70 DM

Anschrift:
MAXON Computer GmbH
KICKSTART PD
Postfach 55 69
6236 Eschborn

Die Diskettenbestellung kann auch telefonisch erfolgen. Der Versand erfolgt dann per Nachnahme.
Tel.: 06196 / 48 18 11
(Mo.-Fr. von 9^{00}-13^{00} u. 14^{00}-17^{00} Uhr)

KICKSTART PD 1-62

PD 62: GRAFIK-DEMOS
HAGENDEMOS: Die Filme, die Joel Hagen zeigt, sind unbestritten Kunstwerke, denn nicht umsonst wurde "RGB" der Gewinner des BADGE KILLER DEMO CONTEST. Die beiden Filme wurden mit "The Director" von "The Right Answers Group" erstellt. (benötigt 1 MB RAM !)
BULLY: Diese Demo ist der absolute Wahnsinn. Drei Bildschirme mit Animationen sind auch noch in Bewegung. Sie gleiten auf und nieder und treten dabei abwechselnd in den Vordergrund. Da zeigt der AMIGA was Multitasking bedeutet.

PD 61: GRAFIK-DEMOS
RIPPLES: eine eindrucksvolle Sequenz, die mit VideoScape 3D erstellt wurde. **WAVEBENCH** bringt Ihren Bildschirm gewaltig ins Wanken; **VIACOM** ist auch nicht viel besser! **DROPSHADOW 2.0:** von nun an hat jedes Fenster eine Schattenseite, mit Einstellregler für Schattenstärke und -länge.

PD 60: RAY-TRACER
A-Render ist ein leitungsfähiges Programm zum Erstellen von Bildern nach dem Ray-Tracing-Verfahren. Dateneingabe per Texteditor. Die Berechnung der Bilder durch das Programm kann, wegen des aufwendigen Verfahrens, mehrere Stunden dauern, aber die Ergebnisse sind hervorragend.

Diskette 59: Grafik-Animation X
Mit drei herrlichen Grafikdemos stellt Eric Graham seine Programme Sculpt 3-D und Animate 3-D vor. Die Bilder der Animationen sind nach dem Ray-Tracer-Verfahren berechnet.

Diskette 58: Grafik-Animation
Die neuesten Demos vom BADGE KILLER DEMO CONTEST. Hervorragende Grafik und Animation.

Diskette 57: UTILITIES
GOMF (bekämpft den GURU), VCHECK (Virustester von Commodore), JOURNAL (Mausaktionen aufnehmen und abspielen), PRINTERSTEALER (Druckerausgabe auf Diskette umleiten), Editordemo

Diskette 56: ASSEMBLER
ASM68K (Macro Assembler mit guter Dokumentation)
ASM (68010 Macro Assembler wie im AmigaDOS Manual beschrieben)
BLINK (bekannter Linker)
AS6502 (portabler 6502-Assembler mit Source in C)

Diskette 55: Grafik/Utilities
Einige schöne Grafikdemos und Utilities zu diesem Thema

Diskette 54: Anwenderprogramme
MICROSPELL (überprüft die Rechtschreibung), ACCESS (Terminal), QBASE (Dateiverwaltung) uvm.

Diskette 53: COMPILER
Auf dieser Diskette befindet sich die Sprache ADL (Adventure Definition Language). Das System besteht aus Compiler, Interpreter und Debugger, wobei alle Teile als Source in C und auch ablauffertig vorliegen.

Diskette 52: UTILITIES
CONMAN V1.0 (sehr nützlicher CLI-Ersatz), MOUSEREADER (Texte lesen mit Mausbedienung), TIMERAM, MEMWATCH, **DISKMAN V1.0** (Bedienung fast aller Diskettenoperationen mit der Maus, sehr umfangreich!), DIRMASTER V1a (sehr schönes Disketten-Verwaltungsprogramm, viele Funktionen)

Diskette 51: C-Compiler
Ein einfacher C-Compiler, in den man etwas Arbeit stecken muß, damit er läuft. Eignet sich für Interessenten am Compilerbau, da der Source-Code vorliegt.

Diskette 50: BASIC
Eine Diskette voll Programmen (Spiele, Grafikprogramme uva.) in AMIGABasic zum Reinschauen, Verändern, Lernen.

Diskette 49: ICONS
Utility-Programme, die sich mit der Erstellung und Manipulation von Icons beschäftigen. z.B.: XICON 2.0 (mit diesem Programm können Dateien ausgeführt werden, die CLI-Kommandos enthalten)

Diskette 48: CRAZY X
Auf dieser Diskette befinden sich nur verrückte Programme, deren Sinn absolut zweifelhaft ist. Allerdings sollten Sie sich diesen Spaß nicht entgehen lassen!

Diskette 47: UTILITIES X
SECTORAMA: ein sehr nützlicher Disketten- und Festplatten-Monitor, mit dem verlorene oder zerstörte Daten wiederhergestellt werden können.
SILICON: sehr komfortabler CLI-Ersatz mit separatem Ausgabe- und Eingabefenster.
DBUG: maschinenunabhängiger Debugger von Fred Fish (Source in C)
TIMER: eine Stoppuhr für die Workbench

Diskette 46: GRAFIK-SHOW X
Eine weitere Diskette (siehe auch PD 42) mit phantastischen RAY-TRACER-Bildern, unterlegt mit digitalisierter, fetziger Musik.

Diskette 45: SPIELE
GRAVITYWARS ist ein interessantes Weltraumspiel, bei dem sich zwei Raumschiffe im Kampf gegenüberstehen.
OTHELLO, eine sehr schöne Reversi-Variante (mit eigenem Fenster!).
STREITPATIENCE, eine Patience-Variante von Hellmut Voelcker (Berlin)
CHESS: spielstarkes Schachprogramm
ADVENTURE: ein Textadventure

Diskette 44: SPIELE X
Adventurefans kennen sicherlich das Grafikadventure HACK (siehe PD 3). Hier die Fortsetzung: LARN. In unüberschaubaren unterirdischen Gängen müssen Gold und Schätze gesucht werden. Aber auch Krafttrunke oder magische Sprüche sollte man nicht ignorieren, denn die benötigt man im Kampf gegen Gnome, Vampire und andere Gestalten. Wirklich sehr empfehlenswert!

Diskette 43: GRAFIK-SHOW
Eine einmalige Show, bei der eine digitalisierte Katze in gleitenden Bewegungen über den Bildschirm trabt. Erstellt wurde diese faszinierende Animation mit einem Digitizer, DPaint und VideoScape 3D.

Diskette 42: GRAFIK-SHOW
Vielfältige nach dem RAY-TRACER-Verfahren erstellte Bilder. Lassen Sie sich von den realistischen Spiegelungen beeindrucken! Mit digitalisierter Musik!

Diskette 41: UTILITIES (Grafik)
Alles, was Sie zu dem von ELECTRONIC ARTS entwickelten Grafik-Standard (IFF-Format) wissen müssen: Laden, Speichern, Komprimieren, Dekomprimieren. Mit Dokumentationen und Source-Codes in C.

Diskette 40: GRAFIK-DEMOS
Boing!, Rotate, Sparks, Moire, Dazzle, 3DCube, Scales, Sizzlers. Sehenswert ist der Film 'Atari meets AMIGA', wer wohl gewinnt? Sehr schön ist das Programm LANDSCAPE, das wunderschöne fraktale Berg- und Tallandschaften erzeugt.

Diskette 39: GRAFIK-SHOW
Stimmungsvolle Landschaftsbilder, die sich gut zum Weiterverarbeiten eignen, und einige digitalisierte Bilder.

Diskette 38: GRAFIK
NoFFP Mandelbrot Set Explorer V2.1 (neue Version) von ABC Softarts in Braunschweig.

Diskette 37: UTILITIES
AddMem (zum Konfigurieren von Speichererweiterungen), MemView (zeigt den Speicherinhalt als Grafik an), GetRom (schreibt das Betriebssystem-ROM des AMIGA 500/2000 als bootfähige Kickstart für den AMIGA 1000 auf Diskette), MegaPatch (paßt die Kickstart des AMIGA 1000 für das autom. Erkennen von internen Speichererweiterungen an)

Diskette 36: CAD
mCAD ist ein wirklich gut gemachtes CAD-Programm, daß jedoch nur im Interlace-Modus läuft. Es bietet die einfachen Zeichenfunktionen und Features wie Zoom, Group, Ungroup, Grid, Move, Rotate). Auf der Diskette sind mehrere Dokumente, die das Programm erklären.

Diskette 35: UTILITIES X
ASDG (resetfeste RAM-Disk), FixDisk, ErrorCk (zur Fehlersuche auf der Diskette). **DiskCat** (erstellt eine Übersicht über die Programme Ihrer Disketten)

Diskette 34: SPIELE
TUNNEL VISION: Werden Sie den Weg durch das Labyrinth finden?
REVERSI: eine spielstarke Version des bekannten Brettspiels
KLONDIKE: ein Patience-Kartenspiel

Diskette 33: GRAFIK-SHOW
Einige sehr gute mit Deluxe Video erstellte Filme. Der benötigte PLAYER ist auch auf der Diskette. INFO: bei AMIGA 500/2000 mit 1MB Speicher erst 'NoFastMem' starten!

Diskette 32: SOUND-DEMOS
Mit einer Demo-Version von SoundScape können digitalisierte Musikstücke abgespielt werden. Erstaunliche Qualität!

Diskette 31: SOUND-DEMO
Dieses Programm erzeugt naturgetreue Geräusche, die über die Tastatur, wie auf einem Klavier, angespielt werden können.

KICKSTART PD-SAMMLUNG:
Die einzelnen Disketten werden nach festen Kriterien zusammengestellt, d.h. daß jede Diskette einen Schwerpunkt hat (z.B. Lehrgänge (Tutor), Bilder-Show, C-Programme, Utilities, Spiele, u. ä.). Außerdem werden Angaben über die Programmiersprache, den verwendeten Interpreter oder Compiler usw. gemacht.

Die Programme laufen auf allen AMIGA-Computern mit Kickstart/Workbench 1.2, allerdings sollten mindestens 512k Speicher vorhanden sein, die meisten Grafikdemos benötigen sogar 1MB Speicher. Sollten dennoch Einschränkungen gelten, wird dies bei den betreffenden Programmen angegeben.

Diskette 30: SOUND-DEMOS
Digitalisierte Songs: Changing Minds, Joan Lui, Miami Vice II, Respactable, Holiday

Diskette 29: UTILITIES
PrfDrvGen (erstellt Drucker-Treiber), DropShadow (jedes Fenster bekommt einen Schatten), MemClear (löscht den Speicher), ScreenSave (speichert den Bildschirm auf Diskette), Compress (komprimiert Programme)

Diskette 28: Editoren
Auf dieser Diskette befinden sich einige schöne Editoren (UEDIT, MED, BLITZ) mit dazugehörigen Zeichensatz-Utilities.

Diskette 26 & 27: Grafik-Show
Auf zwei randvollen Disketten erleben Sie eine einmalige Dia-Show mit hervorragend digitalisierten futuristischen Bildern in voller PAL-Auflösung. Dazu gibt es stimmungsvolle, sphärische Musik.

Diskette 25: UTILITIES
CLOCK, PORTAR, MACView, Kickbench, Disassembler, Tracker, Checkmodem, POPCLI und vieles mehr

Diskette 24: Grafik-Show
Sehr schöne, digitalisierte Frauengesichter.

Diskette 23: Grafik-Show
Viele abwechslungsreiche Motive in verschiedenen Auflösungen, mit Grafik-Show.

Diskette 22: Sprachen
C-FORTH (recht leitungsfähiger FORTH-Interpreter, der auch als Quelltext vorliegt)

Diskette 21: AMIGA-Tutor
Einführung in die Bedienung des AMIGA 500. Ein farbenfroher Lehrgang, der ganz am Anfang beginnt und mit vielen Bildern und Grafiken die Grundbegriffe des AMIGA erklärt. (für Anfänger, komplett in deutsch)

Diskette 20: Grafik-Show
'Fred the Baker und Rose's Flower Shop' COMIC-Film, der die Multitasking Fähigkeiten des AMIGA erklärt

Diskette 19: Grafik-Show
Sehr schöne digitalisierte H.A.M.-Bilder

Diskette 18: Grafik
MANDELBROT-Generator

Diskette 16: Sprachen
XLISP 1.7 (neueste Version) mit ausführlicher Anleitung (über 50k)

Diskette 14: EDITOR
Bekannter Texteditor MICROEMACS Version 30 (viele Features: Search/Replace/Copy)

Diskette 13: Grafik
Sehr schöne Bilder-Show (IFF-Format)

Diskette 12: Grafik
Digitalisierte Bilder mit erstaunlicher Qualität (IFF-Format)

Diskette 11: Grafik-Show X
RAY TRACERS: wunderschöne räumliche Bilder, die auf einer VAX berechnet und auf den AMIGA übertragen wurden

Diskette 10: Grafik-Show X
JUGGLER DEMO: ein bewegliches Männchen jongliert mit drei verspiegelten Kugeln, sehr schöne Demo

Diskette 9: Grafik-Show X
Grafik-Show mit bekannten Cartoons und schönen Landschaftsbildern

Diskette 8: Spiele X
Monopoly: das bekannte Brettspiel mit sehr schöne Grafik, einfache Mausbedienung, bis zu vier Spieler (Source in ABasic)

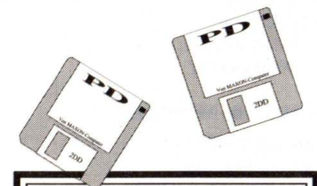
Diskette 7: UTILITIES X
QuickCopy (gutes Kopierprogramm), **DirUtil** (File-Copy), **FileZap** (File-Monitor), **DiskZap** (Disk-Monitor), **DiskSalv** (Diskettenretter), **System-Monitor, CSH** (UNIX-ähnliche Shell)

Diskette 5: Terminal-Programme
WOMBAT (VT102/52 Emulator, XModem, autodial), VT100 (grafikfähig, Soure in C), TermPlus (XModem, Soure in C), DG210 (Data General D-210 Emulator), Ahost (XModem, Kermit), TEK4010 (XModem, VT100)

Diskette 3: Spiele X
HACK: das bekannte Textadventure, das ursprünglich auf UNIX-Rechnern erstellt wurde, liegt hier als spezielle Grafikversion für den Amiga vor.

Diskette 2: Spiele X
YachtC (Würfelspiel für 4 Personen), Puzzle, Missile (verteidigen Sie Ihre Stadt, starker Sound), TriClops (sehr schönes 3D-Spiel), Breakout (3D-Effekt mit Brille), Trek73 (bekannte Star Trek-Variante)

Diskette 1: C-Source
Eine Sammlung von Programmen, die besonders dem Anfänger zeigen, wie man Intuition programmiert. Die Programme liegen sowohl aus C-Quellcode als auch als fertige Programme vor, die sofort gestartet werden können.

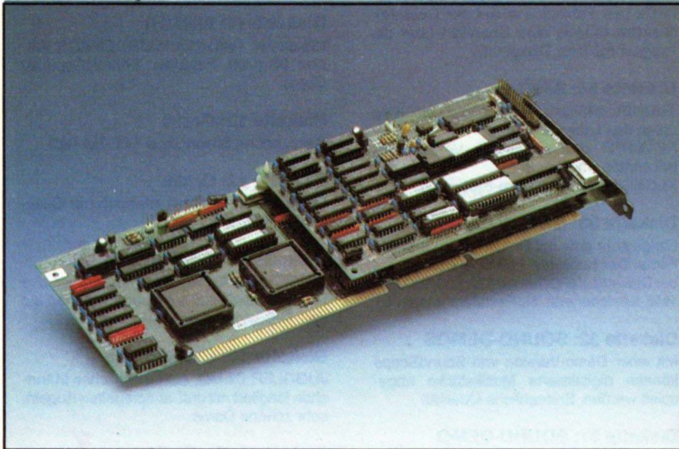


ANRUF GENÜGT
06196 / 481811

VORSCHAU

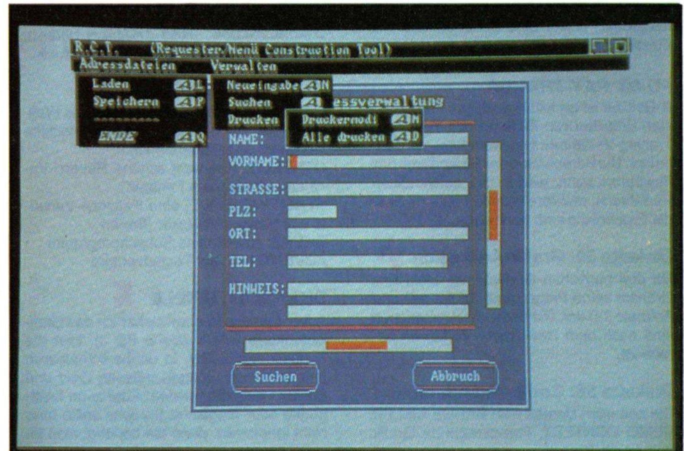
AT-Karte

Die endgültige Version der AT-Karte erreichte unser Haus. Einen Test auf Herz und Nieren lesen Sie in der nächsten KICKSTART. Dort zeigt sich, ob es Probleme mit bestehender Software gibt, ob die Geschwindigkeit der AT-Karte mit anderen ATs gleichzusetzen ist oder ob Schwierigkeiten im Zusammenspiel zwischen AMIGA und Karte auftraten.



R.C.T.-Anwendungen

Die Leistungen des R.C.T. (Requester Construction Tool) sind überwältigend. In der nächsten Ausgabe werden wir Sie mit gezielten Anwendungen in die neue Generation der Requester- und Menüprogrammierung einführen. Die Programmiersprache spielt hierbei keine Rolle, denn das R.C.T. unterstützt alle Sprachen, sei es C, Modula, Assembler, AMIGA- und GFA-BASIC oder Pascal. Sie werden überrascht sein wie schnell und einfach sich mit dem R.C.T. arbeiten läßt.



CLI und Workbench

Einsteiger haben ihre Schwierigkeiten mit den Oberflächen des AMIGA. Auf der einen Seite ist die grafikorientierte Oberfläche - die Workbench, wobei die Intuition-Library als Grundlage dient, auf der anderen Seite ist der Befehlszeileninterpreter - das CLI, der ähnlich der bekannten Betriebssystemoberfläche MS-DOS aufgebaut ist. Was sind die Unterschiede, was die einzelnen Vorteile? Die Beantwortung dieser und anderer Fragen erfahren Sie im nächsten Heft.

Änderungen vorbehalten

Ab 17. Februar an Ihrem Kiosk.

Impressum

KICKSTART

Chefredakteur:
Uwe Bärtels (Chefredakteur)(UB)
Markus Nerding (Stellvertreter) (MN)

Redaktion:
Andreas Krämer (AK)
Harald Schneider (HS)
Marcelo Merino (MM)
Harald Egel (HE)
Christian Keller (CHK)

Herausgeber:
MAXON-Computer GmbH
Industriestraße 26
Postfach 5569
6236 Eschborn
Tel.: 06196/481811
FAX: 06196/41137

Redaktionelle Mitarbeiter:
Gerald Carda (GC)
Jobst Hermeier (JH)
Carsten Borgmeier (CBO)
Martin Silbernagl (MS)
Sven Stillich (SS)
Florian Du Bois (FDB)
Sebastian Dosch (SD)
Wolf Dietrich (WD)

Public Relations:
Claus Peter Lippert

Auslandskorrespondenz:
D. dela Fuente (GB)
L. Hennely (USA)

Verlag:
Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt 13
Tel.:06151/56057
FAX:06151/55689
06151/56059

Verlagsleitung:
Hans-Jörg Heim

Anzeigenverkauf:
Kyriakulla Margaritis
Uwe Heim (Ltg.)

Anzeigenpreise:
nach Preisliste Nr.3, gültig ab 1.1.88

Fotografie, Titelbild:
Klaus Ohlenschläger Foto & Design

Illustrationen:
Barbara Jacobs

Produktion:
Karl-Heinz Hoffmann

Druck:
Ferling Druck, Darmstadt

Bezugsmöglichkeit:
Zeitschriftenhandel, Kauf- und Warenhäuser, Commodore-Fachhändler oder direkt beim Verlag.
KICKSTART erscheint 11 mal im Jahr

Einzelpreis: DM 7,- ,ÖS 56,- SFr 7,-
JahresabonnementInland: DM 70,-
Europ. Ausland DM 90,-
Luftpost DM 120,-

In den Preisen enthalten sind die gesetzliche Mehrwertsteuer und die Zustellgebühren.

Wir bedanken uns bei:
den BLUES BROTHERS - Olaf & Joachim

Die Redaktion

Frischer Wind auf dem Hardwaresektor

ALCOMP
COMPUTERHARDWARE

Profilaufwerk 3,5"
Metallgehäuse ● einstellbare Lauwerknummer mit Displayanzeige ● digitale Trackanzeige ● Write Protect am Laufwerk schaltbar ● abschaltbar ● durchgeschleifter Bus
1 Jahr Garantie

Super ALCOMPreis	329,-

Laufwerk 5,25"
40/80 Track ● Laufwerksbus durchgeschleift ● abschaltbar ● einstellbare Adressen ● MS- DOS- kompatibel ● mit Diskchange

Super ALCOMPreis	298,-
HD 1,6 MB (umschaltbar)	318,-
Amigafarbene Blende	+10,-
Write Protect Schalter	+15,-

Gemischtes Doppel 3,5/5,25"
einzeln ein-/abschaltbar ● einstellbare Laufwerksnummern mit Anzeige ● durchgeschleifter Bus ● bei 5,25" 40/80 Tracks umschaltbar ● Metallgehäuse ● 1 Jahr Garantie

Super ALCOMPreis	548,-

500er Speichererweiterung
Für 512k zusätzliches RAM ● alle RAM-s gesockelt ● selbstkonfigurierend ● abschaltbar ● Uhrenschaltung auf Platine mit Akku- bzw. Batteriepufferung nachrüstbar

Komplett mit 512k	Preis auf Anfrage
Superpreis mit Uhr	Preis auf Anfrage
Bauteilesatz für Uhr ohne Akku	24,-
Leerplatine mit Stecker	*39,-

*mit Schaltplan und Bestückungsliste

Laufwerkanschlußkabel
Zum Anschluß von Laufwerken an alle Amigas ● mit Ansteuerelektronik

Für 3,5" Laufwerk	39,-
Für 5,25" Laufwerk	49,-

Steckplatzerweiterung 3-fach für Laufwerke
Jeder Steckplatz abschaltbar und einstellbare Laufwerksnummer ● Steckplatzerweiterung direkt am Amigagehäuse ● Dadurch keine Kabellängenprobleme

Anschlußfertig zum Super ALCOMPreis	49,-

Soundsampler
Für alle Amiga's mit Software ● Type bei Bestellung bitte angeben ● 8-Bit Datenbreite ● Betrieb am Parallelport (Druckerport) ● Mit Vorverstärker für Micro-Anschluß (Cinch-Buchsen) ● Musik- und Sprachdigitalisierung möglich ● Arbeitet mit fast allen Digitizer-Programmen ● Formschönes Gehäuse

Super ALCOMPreis	79,-

Sampler Studio
● Professionelles Sampler-Programm ● 4-Kanal-Technik ● speichern auf 4 Disketten hintereinander möglich ● alle gängigen Formate (IFF, Data, Future) ● Echtzeitdisplay mit Zoomfunktion ● viele Verfremdungsmöglichkeiten ● Echo, Hall, Reverse

	69,-
Paket: Sampler + Software	129,-

MIDI - Interface
4 Kanäle einschließlich 1 Thru ● Optische Datenanzeige ● Formschönes Gehäuse

Wahnsinnspreis von nur	89,-

Kickstartumschaltung
Bauen Sie die anderen Kickstart-Versionen in Ihren Amiga 500 ● Einfacher Einbau ohne Löten ● für Original-Kickstart-ROM und 2 zusätzliche Versionen auf EPROM ● EPROM-Programmierservice auf Anfrage

SuperALCOMPreis	59.-
Kickstartversion auf EPROM's	120.-

Userport + Experimentierkarte für Expansionport
Mit Lochraster und 2 x 6522 Ports

Leer	59.-
komplett aufgebaut	89.-

Wir suchen ständig Hardware-Entwicklungen. Wir garantieren gute Umsatzprovisionen und ehrliche Abrechnung

ausgereifte Ingenieurleistung ● 14 Tage Umtauschrecht ● fast alle IC's gesockelt ● nur professionelle Leiterplatten ● Bauteile namhafter Hersteller ● mit Bedienungsanleitung



kostenloses Info anfordern!!!

Bestellung und Versand

ALCOMP
A. Lanfermann
Lessing Str. 46
5012 Bedburg
Tel. 0 22 72/15 80

Nachnahmeversand NN-Spesen 7.50 DM b. Vorkasse 3.- DM. Auslandsbestellungen: Nachnahmeversand NN-Spesen 10.- DM b. Vorkasse 5.-DM. Wir liefern Ihnen auf Ihre Rechnung und Gefahr zu den Verkaufs- und Lieferbedingungen des Elektronikgewerbes.
Postgiroamt Köln
(BLZ 370 100 50) 275 54-509





3,5" Laufwerk
Für alle Amiga's ● einstellbare Gerätenummer ● abschaltbar ● Metallgehäuse ● superflach ● 1 Zoll (2,54cm) ● durchgeschleifter Bus ● TEAC Laufwerk
1 Jahr Garantie

komplett anschlußfertig	239,-
Amigafarbene Blende	+10,-

Basislaufwerke
1 Jahr Garantie

TEAC FD 135 FN 3,5" 1MB superslimline	218,-
TEAC FD 55 GFR 5,25" 40/80 Tracks	239,-
Amigafarbene Blende	+10,-
1,6 MB Diskchange	259,-
3,5" Gehäuse	25,-
5,25" Gehäuse	25,-
Gehäuse für "Gemischtes Doppel"	65,-

Bootselector 19,90

Amiga Eprommer
- Für A 500/1000
- Expansionsportanschluß
- Für EPROM's 2764-27011 (8K-128K) Alle A-Typen und CMOS-Typen
- Funktionen:
  LEERTEST — LADEN VON DISK
  VERGLEICHEN — SPEICHERN AUS DISK
  AUSLESEN — HEXDUMP
  BRENNEN
- vier Programmieralgorithmen 50mS/Byte - Superschnell 64K-1,5 min
- Programm zum Generieren und Brennen von Kickstarts direkt von Diskette oder aus ROM
- Mit Software + Gehäuse **225,-**

Meß- und Steuerinterface
- 8 ADC-Kanäle 0-2,55V in 0,01V Stufe
- 1 DAC-Kanäle 0-2,55V in 0,01V Stufe Genauigkeit- 1,5 LSB
- 8 frei programmierbare TTL-I/O Kanäle
- Mit Gehäuse, Anschlüße auf Schraubklemmen
- interne Referenzspannung
- Expansionsanschluß
- Einfache Programmierung in Basic möglich Multitasking tauglich
- incl. DEMO-Software auf 3,5" Diskette

239,-

Trackanzeige
Für DFO-DF3 einstellbar ● für alle Laufwerke (3,5"/5,25") ● Laufwerkbus durchgeschleift ● mit Gehäuse

Super ALCOMPreis	69,-

Einführungsangebot Amiga - Harddisks

HD-Interface A 2000	198,-
HD-Interface A 500/A 1000	249,-
Platte 20 MB A 2000	798,-
30 MB A 2000	898,-
40 MB A 2000	1 098,-
65 MB A 2000	1 348,-
Platte A 500/A 1000 20 MB	898,-
30 MB	998,-
40 MB	1 248,-
65 MB	1 498,-

Vokabeltrainer
2500 englisch-deutsche Vokabeln incl. Hilfssatz ● Merkfunktion ● komfortabler Editor zur Vokabelverwaltung ● Wörterbuch zum Dateiendurchsuchen

59,-



Selbstbootende Harddisk für Amiga ohne PC-Karte!
Die Amiga-Festplatte von ALCOMP:
● Selbstbootend wie "Card" oder "Rad"! ● Als Einbau-Festplatte für den "Amiga 2000" ● Als Externe Einheit für den "Amiga 500" und 1000 mit Gehäuse, eigenem Netzteil und Erweiterungsanschluß ● Erhältlich mit 20, 30, 40 und 65 Megabyte ● Kopiert 1 Megabyte in unter 4 Sekunden ● Speichert schneller als "1.2-Ramdisk" ● Läuft mit "FastFileSystem" ● Einfach einstecken, Formatieren, "Mountlist" und "Startup-Sequence" ändern und los geht's!
Entwickler: Stephan und Stefan
Für den Selbstbau: Harddisk-Interface incl. Steuersoftware ● Anschluß mit Slot für Omti-Controller

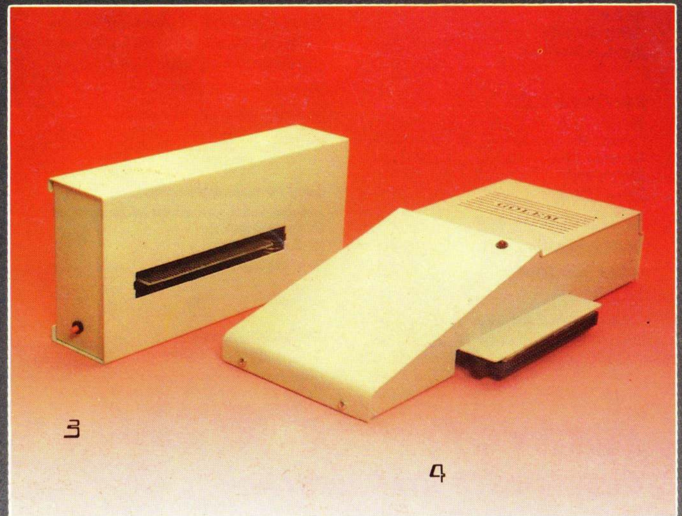
Entwurf + Druck PLUM, Düren-Oberzier, TEL. 02428/711

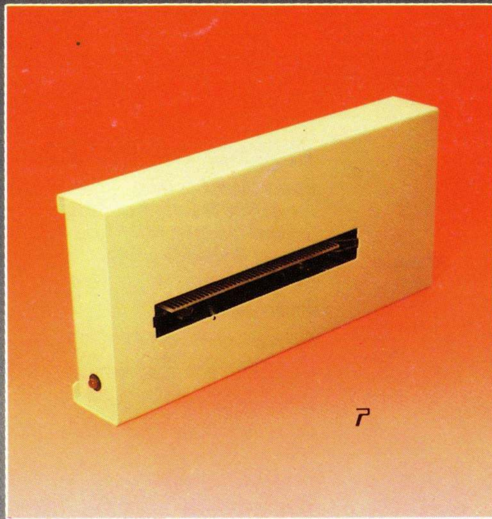
KUPKE

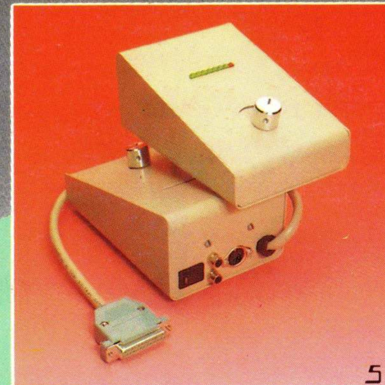


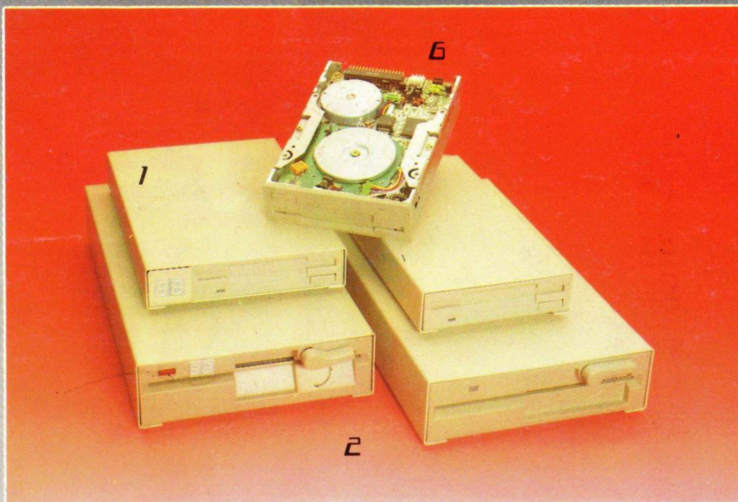
02 31/81 83 25-27
Telefax 02 31/81 74 29
D-4600 Dortmund 1
Burgweg 52 a

GOLEM









Wir liefern im 3-Tage-Rhythmus

1 Golem Drive 3,5 Display
NEC 1037a mit heller Frontblende ● Amiga-farbenes Metallgehäuse ● Abschalter ● Busdurchführung bis DF3 ● Sidecar, PC 1 und PC Karten-kompatibel ● Trackdisplay zur aktuellen Spur- und Kopfanzeige

mit **Display**	**DM 359.-**
ohne **Display**	**DM 339.-**

2 Golem Drive 5,25 Display
NEC Laufwerk mit heller Frontblende ● Amiga-farbenes Metallgehäuse ● Abschalter 40/80 Track-Umschalter ● Busdurchführung bis DF3 ● PC Karten, Sidecar und PC 1 kompatibel ● Trackdisplay zur aktuellen Spur- und Kopfanzeige

mit **Display**	**DM 449.-**
ohne **Display**	**DM 419.-**

3 Golem Ram Box 2MB
2MB Speichererweiterung für dem Amiga 1000 ● ansteckbar am Systembus ● Abschalter ● Busdurchführung ● autokonfigurierend ● Betriebskontrollanzeige ● Amiga-farbenes Metallgehäuse ● erweitert den Grundspeicher auf 2,5 Megabyte

komplett	**DM 1398.-**
ohne Ram's	**DM 449.-**

4 Golem 500 Ram Box
2MB Speichererweiterung im formschönen 500'er Design ● Busdurchführung ● autokonfigurierend ● Betriebskontrollanzeige ● externer Anschluß an den Systembus ● erweitert den Grundspeicher auf 2,5 Megabyte

komplett	**DM 1398.-**
ohne Ram's	**DM 449.-**

Technische Änderungen vorbehalten

5 Golem Sound Stereo
Audio Digitizer der Spitzenklasse ● kompatibel zur meisten Samplersoftware ● DIN- und Cinch Anschluß auch für Micro geeignet ● optisches Aussteuerungsdisplay ● Stereowandlung ● umschaltbar auf Mono-Betrieb

Stereo	**DM 189.-**
Mono ohne Display	**DM 139.-**

6 Golem Drive A 2000
internes Amiga Drive ● NEC 1036a mit heller Frontblende ● einbaufertig modifiziert ● mit Staubschutzklappe ● incl. Einbauanleitung und Montagesatz

DM 200.-

7 Kickstart/Uhrenmodul
"Bitte Workbench einlegen", meldet ihr Amiga 1000 nach dem Einschalten mit dem extern ansteckbaren Kickstartmodul ● Busdurchführung ● Abschalter, so daß andere Kickstartversionen wieder gebootet weden können ● alle gängigen Kickstart-Versionen lieferbar

DM 199.-

Amiga 500/2000 kompatibeles Uhrenmodul ● Akkugepuffert ● extern ansteckbar

im Extragehäuse	**DM 149.-**
Uhr u. Kick in einem Gehäuse	**DM 299.-**

8 Kickstartumschaltplatine
Intern einsteckbare Umschaltplatine bestückt mit einem zusätzlichen Kickstart ● alle gängigen Versionen ● keine Lötarbeiten erforderlich ● umschaltbar auf original Kickstart

komplett	**DM 149.-**
ohne Eprom's	**DM 59.-**

Entwurf + Druck PLUM, Düren-Oberzier, TEL. 024 28/7 11